



# ការអនុវត្តភាសា C

រៀបរៀងដោយ  
មោង សំអាន  
២០២២



**CAMBODIA**  
RESEARCH CREATIVITY INNOVATION

គាំទ្រថវិកាលើការរៀបរៀង និងនិងកែលម្អដោយ  
មូលនិធិស្រាវជ្រាវ គំនិតច្នៃប្រឌិត និងនវានុវត្តន៍



# ការអនុវត្តភាសា C

រៀបរៀងដោយ  
បេន សំអាន  
២០២២



គាំទ្រថវិកាលើការរៀបរៀង និងផ្តល់ដោយ  
មូលនិធិស្រាវជ្រាវ គំនិតច្នៃប្រឌិត និងនវានុវត្តន៍

**សាកលវិទ្យាល័យភូមិន្ទភ្នំពេញ**  
**មហាវិទ្យាល័យវិទ្យាសាស្ត្រ**  
**ដេប៉ាតឺម៉ង់ព័ត៌មានវិទ្យា**

**គណៈកម្មការនីតិវិធី**

លោក ហេង សំអាត

**គណៈកម្មការត្រួតពិនិត្យ**

លោក ជី គ្លង់

លោក ជ ច័ន្ទជារ៉ា

លោក ជូ វណ្ណ

## មុព្វកថា

ដំណើរអភិវឌ្ឍន៍នៃព្រះរាជាណាចក្រកម្ពុជានៅក្នុងយុគសម័យទំនើបនេះ ជាមេរៀនដ៏ជោគជ័យបំផុតមួយដែលចាប់បួសគល់ចេញពីការបញ្ចប់របបប្រល័យពូជសាសន៍ ការបញ្ចប់សង្គ្រាម ការផ្សះផ្សារជាតិ ការកសាងមូលដ្ឋានរឹងមាំនៃសន្តិភាពនិងស្ថេរភាព និងការអភិវឌ្ឍសេដ្ឋកិច្ច។ នៅក្រោយពេលដែលសន្តិភាពត្រូវបានកើតឡើងដោយបរិបូណ៌នៅឆ្នាំ១៩៩៨ កម្ពុជាទទួលបានកំណើនសេដ្ឋកិច្ចខ្ពស់ គឺប្រមាណ៨% ក្នុងមួយឆ្នាំ។ លើសពីនេះទៀត អត្រានៃភាពក្រីក្រត្រូវបានកាត់បន្ថយពីប្រមាណ៥៣% នៅឆ្នាំ២០០៤ មកនៅទាបជាង១០% នៅឆ្នាំ២០១៩។ ដំណើរនៃការអភិវឌ្ឍជាតិជាសកម្មភាពដែលបន្តទៅមុខជាប់ជានិច្ច ហើយគោលនយោបាយថ្មីៗដែលមានលក្ខណៈអន្តរវិស័យគ្របដណ្តប់ ក៏កំពុងលេចរូបរាងឡើង ដើម្បីតម្រង់ទិសកម្ពុជាឆ្ពោះទៅកាន់ប្រទេសមានប្រាក់ចំណូលមធ្យមកម្រិតខ្ពស់នៅឆ្នាំ២០៣០ និងឈានឡើងជាប្រទេសមានប្រាក់ចំណូលខ្ពស់ នៅឆ្នាំ២០៥០។ ការប្រែប្រួលឆាប់រហ័សនៃនិម្មាបនកម្មពិភពលោកនិងតំបន់ រួមទាំងទំនាក់ទំនងភូមិសាស្ត្រនយោបាយ បានផ្តល់កាលានុវត្តភាពសម្រាប់ការអភិវឌ្ឍឧស្សាហកម្មនៅកម្ពុជា ដែលត្រូវបានរាជរដ្ឋាភិបាលចាត់ទុកជាមូលដ្ឋានគ្រឹះនៃកំណើនសេដ្ឋកិច្ចកម្ពុជា។ រាជរដ្ឋាភិបាលកម្ពុជាបាន និងកំពុងបន្តពង្រឹង និងអភិវឌ្ឍវិស័យអប់រំឆ្ពោះទៅរកការស្រាវជ្រាវ និងនវានុវត្តន៍ ដើម្បីពង្រឹងសមត្ថភាពនិងជំនាញរបស់ធនធានមនុស្សនៅកម្ពុជា ឱ្យស្របទៅនឹងបរិបទថ្មីនៃការអភិវឌ្ឍ ជាពិសេសការពង្រឹងសហគ្រិនភាពក្នុងការរៀបចំម៉ូដែលធុរកិច្ចថ្មីៗ។ ដើម្បីចាប់យកកាលានុវត្តភាពពីបដិវត្តន៍ឧស្សាហកម្មទី៤ និងសេដ្ឋកិច្ចឌីជីថលដែលកំពុងផុសផុលឡើង ប្រព័ន្ធអេកូឡូហ្សីដែលបង្កលក្ខណៈអំណោយផលដល់ការបង្កើតថ្មី នវានុវត្តន៍ ការស្រាវជ្រាវ និងអភិវឌ្ឍន៍ ត្រូវតែមានការកែលម្អ។

បណ្តាប្រទេសនៅទ្វីបអាស៊ីកំពុងនាំមុខក្នុងការវិនិយោគលើការស្រាវជ្រាវនិងអភិវឌ្ឍដោយមានភាគហ៊ុនប្រមាណ៤៤% នៃការវិនិយោគទាំងមូលរបស់ពិភពលោក។ ប្រទេសចិនកំពុងបន្តកសាងហេដ្ឋារចនាសម្ព័ន្ធនៃការវិនិយោគលើការស្រាវជ្រាវនិងអភិវឌ្ឍ ក៏ដូចជាសមត្ថភាពមនុស្ស។ ផ្ទុយទៅវិញ ប្រទេសនៅទ្វីបអាមេរិកខាងត្បូង និងអាហ្វ្រិក កំពុងស្ថិតនៅឆ្ងាយពីការវិនិយោគនេះ ហើយជាលទ្ធផល ប្រទេសទាំងនោះក៏ពុំមានកំណើនសេដ្ឋកិច្ចគួរឱ្យកត់សម្គាល់ដែរ។ ទុនវិនិយោគសរុបលើការស្រាវជ្រាវនិងអភិវឌ្ឍរបស់ប្រទេសនៅទ្វីបអាមេរិកខាងត្បូងនិងអាហ្វ្រិក មានប្រមាណ៥%នៃការវិនិយោគ ទាំងមូលរបស់ពិភពលោក ក្នុងពេលដែលតំបន់ទាំង២នេះមានប្រជាជនប្រមាណ២០%នៃប្រជាជនពិភពលោក។ ប្រទេសចំនួន៦ ដែលមានលំដាប់ខ្ពស់ជាងគេក្នុងការវិនិយោគលើការស្រាវជ្រាវនិងអភិវឌ្ឍ រួមមានសហរដ្ឋអាមេរិក ចិន ជប៉ុន អាល្លឺម៉ង់ ឥណ្ឌា និងកូរ៉េខាងត្បូង ដែលស្មើនឹងប្រមាណ ៧០%នៃទុនវិនិយោគសរុបរបស់ពិភពលោក។

តើចំណេះដឹង ផលិតផល និងសេវាកម្មថ្មីទាំងនេះកើតឡើងពីអ្វី? ហើយកើតឡើងដោយរបៀបណា? ព្រះរាជាណាចក្រកម្ពុជាកំពុងតែកសាងមូលដ្ឋានសម្រាប់ការត្រៀមខ្លួនទទួល និងប្រកួតប្រជែងក្នុងយុគសម័យបដិវត្តឧស្សាហកម្មទី៤ នៅក្នុងសេដ្ឋកិច្ចដែលផ្អែកលើពុទ្ធិ ហើយដែលប្រការនេះចាំបាច់តម្រូវឱ្យពលរដ្ឋកម្ពុជា ត្រូវក្លាយខ្លួនជាពលរដ្ឋឌីជីថល ពលរដ្ឋសកល និងពលរដ្ឋដែលប្រកបដោយការទទួលខុស

ត្រូវ ដែលមានសមត្ថភាពក្នុងការផលិត ចែកចាយ និងប្រើប្រាស់ពុទ្ធដើម្បីទទួលបានមនុស្សធន និងរួមចំណែកក្នុង កំណើន។ ធនាគារពិភពលោកបានធ្វើការកត់សម្គាល់តាំងពីឆ្នាំ២០០២នូវបម្លាស់ប្តូរនៃមូលដ្ឋានសេដ្ឋកិច្ច ពី សេដ្ឋកិច្ចដែលពឹងផ្អែកលើកម្លាំងពលកម្ម និងធនធានអតិកម្ម (Labour and Resource Based Economy) ទៅកាន់សេដ្ឋកិច្ចដោយយកពុទ្ធិ (Knowledge Based-Economy) ជាគន្លឹះនៃការអភិវឌ្ឍ។ អាស្រ័យហេតុនេះ នៅលើគន្លងដែលកម្ពុជាកំពុងធ្វើដំណើរឆ្ពោះទៅកាន់សេដ្ឋកិច្ចឌីជីថល សង្គមកម្ពុជាត្រូវតែ មានសមត្ថភាពក្នុងការផលិត ជ្រើសរើស បន្សំ បង្កើតមុខរបរ និងប្រើប្រាស់ពុទ្ធិ ដើម្បីរក្សានិរន្តរភាពនៃ កំណើន និងកែលម្អជីវភាពរស់នៅ។ សមត្ថភាពទាំងនេះអាចកើតឡើងនៅពេលពលរដ្ឋកម្ពុជាមានឱកាស ទទួលបានបទពិសោធពីការស្រាវជ្រាវ ការបណ្តុះគំនិតច្នៃប្រឌិត និងការស្វែងរកនវានុវត្តន៍។

កំណែទម្រង់វិស័យអប់រំ គឺជាការត្រួសត្រាយមាត់សម្រាប់ដំណើរឆ្ពោះទៅកាន់សង្គមប្រកបដោយពុទ្ធិ និងប្រជាពលរដ្ឋប្រកបដោយភាពរស់រវើក។ តាមរយៈមូលដ្ឋានអប់រំ សង្គមប្រកបដោយពុទ្ធិនឹងប្រមូលផ្តុំ បង្កើត និងចែករំលែក ទៅកាន់សមាជិកក្នុងសង្គមនូវសម្បទាអប់រំ ពិសេសគឺពុទ្ធិសម្បទា ក្នុងបុព្វហេតុនៃ មនុស្សជាតិនិងឧត្តមប្រយោជន៍នៃប្រទេស។ សង្គមប្រកបដោយពុទ្ធិ គឺពុំគ្រាន់តែជាសង្គមដែលសម្បូរព័ត៌មានប៉ុណ្ណោះទេ តែជាសង្គមដែលប្រជាពលរដ្ឋអាចធ្វើបរិវត្តកម្មពីព័ត៌មានទៅជា មូលធនប្រកបដោយប្រសិទ្ធភាព។ ការរីកចម្រើនទៅមុខជាលំដាប់នៃបច្ចេកវិទ្យានិងគំណាត់ បានពង្រីកព្រំដែននៃការចូលទៅកាន់ និង ការទទួលបានព័ត៌មានជាសកល ហើយដែលក្នុងន័យនេះ ការអប់រំនឹងបន្តវិវត្តទៅមុខនិងមានការផ្លាស់ប្តូរ។ សង្គមមួយដែលមានអំណាន និងរបាប់ជាបុរេលក្ខខណ្ឌនៃជីវភាពប្រចាំថ្ងៃនៃប្រជាពលរដ្ឋ ពេលនោះបំណិន នៃអំណាន និពន្ធ និងការគណនាលេខនព្វន្ត គឺជាចលករនៃការរៀនរបស់សិស្ស។ ធាតុដ៏ចម្បងមួយដែលស្ថិត នៅក្នុងការកសាងសង្គមដែលប្រកបដោយពុទ្ធិគឺសៀវភៅសិក្សា ហើយការរៀបរៀង និពន្ធ និងកែលម្អសៀវ ភៅសិក្សាជាប្រចាំគឺជានវានុវត្តន៍នៃវិស័យអប់រំដែលនាំទៅរកការសិក្សាពេញមួយជីវិត ការអភិវឌ្ឍសម្បទាអប់រំ និងការចែករំលែកចំណេះដឹង។ មូលដ្ឋានអប់រំ ជាពិសេស គឺគ្រឹះស្ថានឧត្តមសិក្សាត្រូវមានតួនាទីដែល ប្រកបដោយការឆ្លើយតបចំពោះតម្រូវការខាងលើនេះ។ សាស្ត្រាចារ្យ អ្នកស្រាវជ្រាវ និងបុគ្គលិកអប់រំត្រូវបន្ត សិក្សាជាប់ជានិច្ច តាមរយៈការរៀបរៀង និពន្ធ និងកែលម្អសៀវភៅសិក្សា ហើយដែលសៀវភៅសិក្សាទាំងនេះ នឹងក្លាយជា ស្ថាននៃទំនាក់ទំនងរវាងនវានុវត្តន៍នៃបច្ចេកវិទ្យា និងការរៀននិងបង្រៀននៅក្នុងថ្នាក់រៀន។

សង្គម ដែលប្រកបពុទ្ធិ ក៏ជាសង្គមដែលបណ្តុះឱ្យមានរចនាសម្ព័ន្ធទន់នៃសេដ្ឋកិច្ចដែលពឹងផ្អែកលើ ពុទ្ធិដែរ។ ឧទាហរណ៍ជាក់ស្តែងនៃបែបផែននេះរួមមាន Silicon Valley នៃសហរដ្ឋអាមេរិក សួនឧស្សាហកម្ម វិទ្យាសាស្ត្រអាកាសយានយន្តនិងយានយន្តនៅទីក្រុង Munich ប្រទេសអាល្លឺម៉ង់ តំបន់ជីវបច្ចេកវិទ្យានៅក្រុង Hyderabad ប្រទេសឥណ្ឌា តំបន់ផលិតគ្រឿងអេឡិចត្រូនិកនិងសារគមនាគមន៍ ឌីជីថលនៅទីក្រុង Seoul ប្រទេសកូរ៉េខាងត្បូង ក៏ដូចជាសួនឧស្សាហកម្មថាមពល និងឥន្ធនគីមីសាស្ត្រនៃប្រទេសប្រេស៊ីល ហើយក៏ នៅមានទីក្រុងនៃប្រទេសជាច្រើនទៀតនៅលើពិភពលោក។ លក្ខណៈសម្បត្តិនៃទីក្រុងទាំងនេះគឺការប្រើ ប្រាស់និន្នាការនៃការអភិវឌ្ឍដែលជំរុញ និងតម្រង់ទិសដោយចំណេះដឹង ហើយដែលចំណេះដឹងទាំងនោះ កើតចេញជាដំបូងពីការវិនិយោគទៅលើគ្រឹះស្ថានឧត្តមសិក្សា ស្ថាប័នស្រាវជ្រាវ មជ្ឈមណ្ឌលឧត្តមភាពនៃ

ជំនាញជាន់ខ្ពស់ ការប្រកួតប្រជែងដោយគុណាធិបតេយ្យ និង ជាពិសេស ការបណ្តុះវប្បធម៌អំណាននិងនិពន្ធ សៀវភៅ។ ល្បឿននៃការរីកចម្រើនផ្នែកពុទ្ធិ និងបច្ចេកវិទ្យាកំពុងមានសន្ទុះលឿនជាងអ្វីដែលសិស្ស និង និស្សិតអាចទទួលបានពីគ្រូនៅគ្រឹះស្ថានសិក្សា ដែលធ្វើឱ្យគោលដៅនៃការអប់រំនៅពេលបច្ចុប្បន្ននេះ មានការ ប្រឈមខ្លាំងជាងពេលណាទាំងអស់។ ឧទាហរណ៍ ក្នុងមួយឆ្នាំ មានសៀវភៅជាង ២,២លានចំណងជើង ត្រូវ បានសរសេរនិងបោះពុម្ព ដែលក្នុងនោះប្រទេសចិនមាន ៤៤០ពាន់ ចំណែកឯសហរដ្ឋអាមេរិកមាន ៣០៥ ពាន់ និងប្រទេសរុស្ស៊ីមាន ១២០ពាន់ចំណងជើង។

ខណៈពេលដែលបច្ចេកវិទ្យាកំពុងរីកចម្រើនជារៀងរាល់ថ្ងៃ មធ្យោបាយសម្រាប់អំណានក៏មានច្រើន ជម្រើសសម្រាប់សិស្ស-និស្សិត និងសាធារណៈជន រួមមានការអានសៀវភៅ ការអានលើឧបករណ៍ អេឡិចត្រូ និក ការអានដោយប្រើទូរសព្ទវៃឆ្លាត និងការអានលើកុំព្យូទ័រ ដែលសុទ្ធសឹងជាមធ្យោបាយសំខាន់ៗដែលនាំ អ្នកអានទាំងឡាយឱ្យសម្រេចគោលបំណងអានរបស់ខ្លួន។ ម្យ៉ាងវិញទៀត អំណានដោយប្រើមធ្យោបាយប ច្ចេកវិទ្យាទំនើប ចំណាយពេលតិច ងាយស្រួលអាន និងជួយដល់បរិស្ថានមួយកម្រិតទៀត។ នាពេលបច្ចុប្បន្ន សិស្ស-និស្សិត និងសាធារណៈជនកម្ពុជាដែលស្រឡាញ់អំណានកំពុងតែប្រើប្រាស់មធ្យោបាយអំណានទាំង នេះ។ បើយើងក្រឡេកមើលទៅប្រទេសជឿនលឿន ទោះបីជាបច្ចេកវិទ្យារីកចម្រើនខ្លាំងយ៉ាងណា អំណាន តាមរយៈសៀវភៅនៅតែមានសន្ទុះដដែល។ ម្យ៉ាងវិញទៀត បច្ចេកវិទ្យាអានបែបទំនើបតាមរយៈឧបករណ៍ ទំនើប អាស្រ័យលើលទ្ធភាពនៃធនធានអប់រំឌីជីថល និងមាតិកាឌីជីថលគ្រប់គ្រាន់ដែលបានផលិត និង បង្ហាញចែកចាយសម្រាប់អំណាន។

ក្នុងបរិបទកម្ពុជា ជាពិសេសក្នុងបរិការណ៍នៃការផ្ទុះរីករាលដាលនៃជំងឺកូវីដ-១៩ ក្រសួងអប់រំ យុវជន និងកីឡា បានជំរុញឱ្យមានបរិវត្តកម្មឌីជីថលនៅក្នុងអេកូស៊ីស្តែមនៃការអប់រំ ជាពិសេស ការអប់រំតាមប្រព័ន្ធអេ ឡិចត្រូនិក និងការអប់រំពីចម្ងាយដើម្បីលើកកម្ពស់អំណានតាមរយៈការផលិតមាតិកាឌីជីថល ដែលមានភាព សម្បូរបែប ការកសាងសមត្ថភាពផ្នែកតំណភ្ជាប់និងវេទិកាឌីជីថល ការពង្រីកវិសាលភាពនៃមជ្ឈមណ្ឌល ទិន្នន័យ និងការលើកកម្ពស់គុណភាពនៃការផលិតធនធានអប់រំឌីជីថល គួបផ្សំជាមួយការចែកសន្លឹកកិច្ចការ ឱ្យសិស្សយកទៅរៀននៅផ្ទះ និងការចុះទៅជួបជាមួយសិស្សជាបណ្តុំនៅតាមសហគមន៍។ ក្នុងន័យលើក កម្ពស់អំណាន និងភាពសម្បូរបែបនៃធនធានសៀវភៅសិក្សា ឱ្យកាន់តែមានប្រសិទ្ធភាពនិងភាពសក្តិសិទ្ធិ និង ផ្តល់ឱកាសអំណានកាន់តែច្រើនថែមទៀតដល់សិស្សានុសិស្ស និស្សិត និងសាធារណៈជន ក្រសួងអប់រំ យុ វជន និងកីឡាលើកទឹកចិត្តនូវចំណុចមួយចំនួនដូចខាងក្រោម៖

1. សាស្ត្រាចារ្យ អ្នកស្រាវជ្រាវ និងបុគ្គលិកអប់រំ សូមបន្តនិងបង្កើនការបោះពុម្ពស្នាដៃបន្ថែមទៀត ដើម្បីធ្វើឱ្យធនធានសម្រាប់អំណានកាន់តែសម្បូរបែប ជាពិសេសធនធានអំណានជាខេមរភាសា
2. គ្រឹះស្ថានឧត្តមសិក្សា សូមផ្តល់លទ្ធភាពគ្រប់បែបយ៉ាង ដើម្បីឱ្យបុគ្គលិកអប់រំគ្រប់លំដាប់ថ្នាក់ និង និស្សិតគ្រប់កម្រិតសិក្សាអាចចូលរួមអាន និងសិក្សាស្រាវជ្រាវតាមគ្រប់លទ្ធភាពជាមួយធនធាន អំណាន ជាពិសេសការរៀបចំឱ្យមានពេលវេលាសម្រាប់សហសិក្សា និងអំណានក្នុងបណ្ណាល័យ

3. សាស្ត្រាចារ្យតាមមុខវិជ្ជា និងអ្នកស្រាវជ្រាវតាមជំនាញឬវិស័យ ត្រូវរៀបចំដំណើរការរៀន បង្រៀន និងស្រាវជ្រាវដែលមានដាក់បញ្ចូលកិច្ចការស្វ័យសិក្សា សហសិក្សា ឬការស្រាវជ្រាវបណ្តាលយ ដែលតម្រូវឱ្យនិស្សិត ត្រូវអាននិងស្រាវជ្រាវជាមួយធនធានអំណាន
4. គ្រឹះស្ថានឧត្តមសិក្សា និងមជ្ឈមណ្ឌលស្រាវជ្រាវ ត្រូវខិតខំឱ្យអស់លទ្ធភាពក្នុងការបង្កើតបណ្តាល យ មជ្ឈមណ្ឌលរក្សាឯកសារ ឬមជ្ឈមណ្ឌលអប់រំឌីជីថល ជាដើម ដើម្បីឱ្យបុគ្គលិកអប់រំគ្រប់ លំដាប់ថ្នាក់និងនិស្សិតគ្រប់កម្រិតសិក្សា អាចទទួលបាន និងស្វែងរកប្រភពសម្រាប់អំណាន កាន់តែសម្បូរបែប និងមានភាពបត់បែន ឆ្លើយតបតាមតម្រូវការអ្នកអាន
5. និស្សិតគ្រប់កម្រិតសិក្សា ត្រូវខិតខំនិងចំណាយពេលវេលាអាន និងចាត់ទុកវប្បធម៌ និងអកប្បកិរិយា អំណានជាផ្នែកមួយ នៃពេលវេលានិងភាពស៊ីវិល័យនៃជីវិតប្រចាំថ្ងៃ
6. បងប្អូនជនរួមជាតិ ដែលជាមាតាបិតា ឬអ្នកអាណាព្យាបាល សូមជួយជំរុញនិងបង្កលក្ខណៈ កាន់តែ ច្រើនថែមទៀត ជាពិសេស ការលែងកំណាយនៅក្នុងគ្រួសារសម្រាប់ការទិញសម្ភារៈ សិក្សា សៀវភៅអាន និងឧបករណ៍សម្រាប់អំណានដល់កូនៗ ដែលចាត់ទុកជាការវិនិយោគមួយ ដ៏សំខាន់ សម្រាប់ បង្កើនចំណេះដឹង និងអនាគតរបស់ពួកគេ។

ដោយមានការគាំទ្រពីក្រសួងសេដ្ឋកិច្ច និងហិរញ្ញវត្ថុ នៅឆ្នាំ២០២០ ក្រសួងអប់រំ យុវជន និងកីឡា បានបង្កើតមូលនិធិស្រាវជ្រាវ គំនិតច្នៃប្រឌិត និងនវានុវត្តន៍ ដែលហៅកាត់ថា “មូលនិធិ ស.គ.ន” និងហៅជា ភាសាអង់គ្លេសថា The Research Creativity and Innovation Fund ដែលហៅកាត់ជាភាសាអង់គ្លេសថា “RCI Fund”។ គោលដៅចម្បងនៃមូលនិធិនេះ គឺរួមចំណែកលើកកម្ពស់វប្បធម៌នៃការស្រាវជ្រាវ បំផុសគំនិត ច្នៃប្រឌិត និងជំរុញការធ្វើនវានុវត្តន៍ ដើម្បីជាប្រយោជន៍ដល់វិស័យអប់រំ យុវជន និងកីឡា ដែលឆ្លើយតបទៅនឹងទី ផ្សារពលកម្ម និងសាកលភាវូបនីយកម្ម។ មូលនិធិ ស.គ.ន បានសម្រេចកំណត់ប្រធានបទ ជាអាទិភាព សម្រាប់គាំទ្រដោយមូលនិធិចំនួន៣ រួមមាន ឌីជីថលនីយកម្មសម្រាប់បដិវត្តឧស្សាហកម្ម៤.០ (Digitaliza - tion for IR.4.0) ការស្រាវជ្រាវអនុវត្តលើវិស័យកសិកម្ម (Applied Agricultural Research) និងការ ស្រាវជ្រាវគរុកោសល្យសតវត្សទី២១ (21st Century Pedagogy Research) ។

ដោយមានការធ្វើអាទិភាពរូបនីយកម្មទៅលើទិសដៅនៃការប្រើប្រាស់ថវិកាមូលនិធិសម្រាប់ ឆ្នាំ២០២០ ក្រសួងសេដ្ឋកិច្ច និងហិរញ្ញវត្ថុ និងក្រសួងអប់រំ យុវជន និងកីឡា បានផ្តល់ការគាំទ្រដល់ការ រៀបរៀង និងនិពន្ធ និងកែលម្អ សៀវភៅសិក្សា(Text book) ដែលនឹងត្រូវប្រើប្រាស់នៅកម្រិតឧត្តមសិក្សា។ គោលបំណងនៃការរៀបរៀង និងនិពន្ធ និងកែលម្អ សៀវភៅសិក្សានៅកម្រិតឧត្តមសិក្សា គឺដើម្បីបង្កើនបរិមាណ លើកកម្ពស់គុណភាព និងពង្រីកសមធម៌នៃធនធានសិក្សាជាខេមរភាសា ជូនដល់និស្សិតដែលកំពុងបន្តការ សិក្សា និងត្រៀមខ្លួនធ្វើការស្រាវជ្រាវនៅកម្រិតឧត្តមសិក្សា។ លើសពីនេះទៀត ការរៀបរៀង និងកែ លម្អសៀវភៅសិក្សានៅកម្រិតឧត្តមសិក្សា មានគោលដៅដូចខាងក្រោម ៖

- ឆ្លើយតបជាបន្ទាន់ចំពោះការខ្វះខាតធនធានសិក្សា ដែលជាតម្រូវការសិក្សារបស់និស្សិតនៅ កម្រិតឧត្តមសិក្សា

- លើកកម្ពស់ទំនើបការបន្ថែម និងឧត្តមានវត្តមាននៃការរៀននិងបង្រៀន និងការស្រាវជ្រាវនៅ លើមុខវិជ្ជា កម្មវិធីសិក្សា ឬមុខជំនាញជាក់លាក់
- បង្កើនភាពស៊ីជម្រៅក្នុងការកសាងវិជ្ជាជីវៈនិងបទពិសោធន៍សម្រាប់ឋានៈសាស្ត្រាចារ្យ និងអ្នក ស្រាវជ្រាវ
- រួមចំណែកដល់ការកសាងភាពជាសហគមន៍វិជ្ជាជីវៈ ការចែករំលែកបទពិសោធន៍ និងវប្បធម៌នៃ ការរៀបរៀង និងនិពន្ធ និងកែលម្អសៀវភៅសិក្សានៅកម្រិតឧត្តមសិក្សា ។

ក្រសួងអប់រំ យុវជន និងកីឡា បានវាយតម្លៃខ្ពស់ចំពោះការបោះជំហានប្រកបដោយមនសិការវិជ្ជាជីវៈ នៃគ្រឹះស្ថានឧត្តមសិក្សា និងបុគ្គលិកអប់រំទាំងអស់ ក្នុងការរៀបចំ រៀបរៀង និងនិពន្ធ និងកែលម្អសៀវភៅសិក្សា ដើម្បីបង្កើនបរិមាណ លើកកម្ពស់គុណភាព និងពង្រឹងសមធម៌នៃធនធានសិក្សាជាខេមរភាសា ជូននិស្សិត ដែលកំពុងបន្តការសិក្សា និងត្រៀមខ្លួនធ្វើការស្រាវជ្រាវនៅកម្រិតឧត្តមសិក្សា ។ សៀវភៅសិក្សាជាផ្នែកមួយនៃ ការទទួលស្គាល់គុណភាពអប់រំនៃគ្រឹះស្ថានឧត្តមសិក្សា និងជាធនធានសិក្សាដែលជាមូលដ្ឋានមួយដ៏សំខាន់ ក្នុងការគាំទ្រដល់ការបង្រៀន និងរៀន ហើយត្រូវមានបរិមាណគ្រប់គ្រាន់ ឆ្លើយតបទៅនឹងកម្មវិធីអប់រំ និង តម្រូវការសិក្សាស្រាវជ្រាវ។ ជាគោលការណ៍ គ្រឹះស្ថានឧត្តមសិក្សាទាំងអស់ ត្រូវមានសៀវភៅសិក្សាដែលប្រើ ជាគោលសម្រាប់មុខវិជ្ជានីមួយៗ។ ចំនួនសៀវភៅសិក្សាដែលគ្រប់គ្រាន់សម្រាប់ការស្រាវជ្រាវ និងការសិក្សា របស់និស្សិត ត្រូវមានយ៉ាងតិចមួយចំណងជើងក្នុងមួយមុខវិជ្ជា ហើយត្រូវតម្កល់យ៉ាងតិច២ច្បាប់ នៅក្នុង បណ្ណាល័យ ឬអាចរកបានតាមប្រព័ន្ធអេឡិចត្រូនិក។ ក្រសួងអប់រំ យុវជន និងកីឡា លើកទឹកចិត្តបន្ថែមទៀត ជូនដល់គ្រឹះស្ថានឧត្តមសិក្សារដ្ឋ និងឯកជនដែលបានស្នើសុំថវិកាមូលនិធិរួច សូមចូលរួមបន្ថែមទៀតដើម្បី បង្កើនចំនួនចំណងជើងសៀវភៅ។ ចំណែកគ្រឹះស្ថានឧត្តមសិក្សារដ្ឋនិងឯកជនដែលពុំទាន់បានដាក់ពាក្យស្នើ សុំ សូមចូលរួមដើម្បីជាគុណប្រយោជន៍ដល់តម្រូវការដ៏ទទួល និងថ្លៃថ្នានៃនិស្សិតកម្ពុជាក្នុងការសិក្សា និង ស្រាវជ្រាវនៅកម្រិតឧត្តមសិក្សា។



## **សេចក្តីបញ្ជាក់**

### **របស់មូលនិធិការស្រាវជ្រាវ គំនិតច្នៃប្រឌិត និងនវានុវត្តន៍**

សៀវភៅសិក្សានេះជាលទ្ធផលនៃការស្នើសុំអនុវត្តវិកាមូលនិធិការស្រាវជ្រាវ គំនិតច្នៃប្រឌិត និងនវានុវត្តន៍ ក្នុងគម្រោងរៀបរៀង និងត្រួតពិនិត្យ សៀវភៅសិក្សា ដែលនឹងត្រូវប្រើប្រាស់នៅកម្រិត ឧត្តមសិក្សា។ សៀវភៅសិក្សានេះ ត្រូវបានរៀបរៀង និងត្រួតពិនិត្យដោយមានការធានាអះអាងថាជាស្នាដៃរបស់អ្នកនិពន្ធផ្ទាល់ និងបានឆ្លងកាត់ត្រួតពិនិត្យ ផ្តល់យោបល់ និងវាយតម្លៃដោយក្រុមប្រឹក្សាអប់រំ ក្រុមប្រឹក្សាស្រាវជ្រាវ ឬក្រុមប្រឹក្សាដែលមានតម្លៃស្មើនៃគ្រឹះស្ថានឧត្តមសិក្សា និងតាមរយៈកិច្ចសន្យាដែលបានធ្វើឡើង និងដែលបានតម្កល់ទុកនៅមូលនិធិការស្រាវជ្រាវ គំនិតច្នៃប្រឌិត និងនវានុវត្តន៍។ រាល់ខ្លឹមសារ ការបកស្រាយ និងរូបភាព គឺជាជំហរនិងទស្សនៈផ្ទាល់របស់អ្នកនិពន្ធ ហើយ ពុំឆ្លុះបញ្ចាំង ឬជាតំណាងដល់មូលនិធិការស្រាវជ្រាវ គំនិតច្នៃប្រឌិត និងនវានុវត្តន៍នៃក្រសួងអប់រំ យុវជន និងកីឡា ឡើយ។

## **អារម្ភកថា**

សៀវភៅនេះត្រូវបានរៀបរៀងឡើងក្រោមការឧបត្ថម្ភគាំទ្ររបស់មូលនិធិស្រាវជ្រាវ គំនិតផ្តួចផ្តើមប្រឌិត និង នវានុវត្តន៍សម្រាប់ទុកជាសៀវភៅសិក្សាគោលដើម្បីបម្រើឱ្យការរៀននិងបង្រៀន “ភាសា C” ក្នុងដេប៉ាតឺម៉ង់ ព័ត៌មាននៃមហាវិទ្យាល័យវិទ្យាសាស្ត្រនៅសាកលវិទ្យាល័យភូមិន្ទភ្នំពេញ។ សៀវភៅនេះផ្តល់ជូនអ្នកសិក្សា ទាំងផ្នែកទ្រឹស្តី និង ជាពិសេស ផ្តោតលើផ្នែកអនុវត្តន៍តាំងពីឆ្នាំ១៩៧០ រហូតដល់បច្ចុប្បន្ន ដែលអាចសម្រួល ឱ្យអ្នកសិក្សាព័ត៌មានវិទ្យាទាំងឡាយមានមូលដ្ឋានគ្រឹះរឹងមាំក្នុងការសិក្សាបន្ត ឬស្រាវជ្រាវបន្ថែម ឬក្នុងការ ប្រកបការងារផ្សេងៗក្នុងវិជ្ជាជីវៈរបស់ខ្លួន។ មិនតែប៉ុណ្ណោះ សៀវភៅនេះក៏អាចចាត់ទុកជាឯកសារពិគ្រោះ សម្រាប់អ្នកស្នេហាព័ត៌មានវិទ្យាផងដែរ។

# សេចក្តីថ្លែងអំណរគុណ

ខ្ញុំបាទ ហេង សំអាន ជាសាស្ត្រាចារ្យដេប៉ាតឺម៉ង់ព័ត៌មានវិទ្យានៃមហាវិទ្យាល័យវិទ្យាសាស្ត្រនៅសាកលវិទ្យាល័យភូមិន្ទភ្នំពេញ និងជាអ្នករៀបរៀងសៀវភៅសិក្សាដែលមានចំណងជើងថា “ការអនុវត្តកាសា C”។ ខ្ញុំបាទសូមសម្តែងនូវការរីកចម្រើនខ្ពង់ខ្ពស់និងសូមគោរពថ្លែងអំណរគុណ និងជឿជាក់យ៉ាងជ្រាលជ្រៅចំពោះ ៖

- ក្រសួងសេដ្ឋកិច្ច និងហិរញ្ញវត្ថុ ដែលបានផ្តល់ថវិកាមកក្រសួងអប់រំ ដើម្បីបង្កើតជាមូលនិធិសម្រាប់គាំទ្រដល់ការសរសេរសៀវភៅសិក្សាសម្រាប់កម្រិតឧត្តមសិក្សារបស់ខ្ញុំបាទនេះ ។

- ក្រសួងអប់រំ យុវជន និងកីឡា ដែលមាន ឯកឧត្តមបណ្ឌិតសភាចារ្យរដ្ឋមន្ត្រី ហង់ជួន ណារ៉ុន ជាប្រមុខ ដែលលោកបានដឹកនាំបង្កើតមូលនិធិស្រាវជ្រាវ គំនិតឆ្នៃប្រឌិត និងនវានុវត្តន៍ឡើង ដើម្បីគាំទ្រទាំងស្រុងដល់ការសរសេរសៀវភៅនេះ ដែលជាចំណែកមួយក្នុងបេសកកម្មកំណែទម្រង់វិស័យអប់រំឧត្តមសិក្សារបស់លោក ។

- ឯកឧត្តមបណ្ឌិត សាន វឌ្ឍនា អនុរដ្ឋលេខាធិការក្រសួងអប់រំ យុវជន និងកីឡា ដែលក្នុងនាមលោកជាតំណាងមូលនិធិ លោកបានយកអស់កម្លាំងកាយចិត្ត ជួយចាត់ចែងនិងជ្រោមជ្រែងយ៉ាងពេញទំហឹង ព្រមទាំងផ្តល់អនុសាសន៍និងដំណោះស្រាយនានា ដើម្បីជំនះនូវរាល់ឧបសគ្គដែលកើតមានក្នុងដំណើរការសរសេរសៀវភៅនេះ ។

- ឯកឧត្តមបណ្ឌិត ជេត ជាលី សាកលវិទ្យាធិការនៃសាកលវិទ្យាល័យភូមិន្ទភ្នំពេញ ដែលបានយកអស់កម្លាំងកាយចិត្ត ព្រមទាំងពេលវេលាដ៏មមាញឹកនិងមានតម្លៃរបស់លោកក្នុងការជួយលើកទឹកចិត្ត ជួយខ្លះខ្លែងដោះស្រាយ និងផ្តល់អនុសាសន៍ដ៏ត្រឹមត្រូវ ក៏ដូចជាបង្គុលក្នុងការងាយស្រួលនានា ទាំងផ្នែករដ្ឋបាល និងទាំងផ្នែកបច្ចេកទេស ដើម្បីឲ្យដំណើរការរៀបរៀងសៀវភៅនេះបានប្រព្រឹត្តទៅដោយរលូនរហូតចប់ចុងចប់ដើម ។

- លោក ឃុំ វ៉ាន់ហ្វា ប្រធានរដ្ឋបាលនិងហិរញ្ញវត្ថុនៃមូលនិធិស្រាវជ្រាវ គំនិតឆ្នៃប្រឌិត និងនវានុវត្តន៍ និងជាប្រធានរដ្ឋបាលនិងហិរញ្ញវត្ថុនៃមជ្ឈមណ្ឌលសម្រាប់ការអប់រំឌីជីថល ដែលបានយកអស់កម្លាំងកាយចិត្តជួយឈឺឆ្កាលក្នុងការដោះស្រាយបញ្ហាប្រឈមនានាយ៉ាងសស្រាក់សស្រាំជាទីបំផុត ដោយមិនគួញក្តីរន្ធត់និងមិនខ្លាចនឿយហត់ ដើម្បីឱ្យការសរសេរសៀវភៅនេះប្រព្រឹត្តទៅដោយរលូន ។

- ឯកឧត្តមបណ្ឌិត ស្រី ថាវិទ្ធ អ្នកសម្របសម្រួលសាលាក្រោយឧត្តម និង លោកបណ្ឌិត សុខ សិរីអនុប្រធានការិយាល័យស្រាវជ្រាវ ដែលតែងជួយឈឺឆ្កាល ជួយសម្របសម្រួល និងផ្តល់សេវាសម្រាប់ការប្រជុំបច្ចេកទេសនានាតាមអនឡាញតាំងពីដើមរៀងមក ។

- អស់លោក-លោកស្រី ជាសមាជិក-សមាជិកាគណៈកម្មការអភិវឌ្ឍសៀវភៅសិក្សា ក៏ដូចជាគណៈកម្មការត្រួតពិនិត្យសៀវភៅសិក្សារបស់សាកលវិទ្យាល័យភូមិន្ទភ្នំពេញ ដែលបានជួយផ្តល់ធាតុចូលដល់សៀវភៅនេះឱ្យកាន់តែមានគុណភាពល្អប្រសើរថែមទៀត ។

- ជាចុងក្រោយ និងជាពិសេស លោក ធួ វេណូ ប្រធានការិយាល័យស្រាវជ្រាវនៃសាកលវិទ្យាល័យភូមិន្ទភ្នំពេញ ដែលតែងផ្តល់ធាតុចូលចំពោះរាល់ឯកសារការងារ ទាំងផ្នែករដ្ឋបាលនិងទាំងផ្នែកបច្ចេកទេសលោកថែមទាំងជួយសម្រួលរាល់បញ្ហាប្រឈមនានាយ៉ាងជិតដល់បំផុត ដោយមិនប្រកាន់ពេលវេលា និងដោយចិត្តជ្រះថ្លា រហូតការសរសេរសៀវភៅនេះទទួលបានជោគជ័យជាស្ថាពរ ។

# មាតិកា

|                                                                  |      |
|------------------------------------------------------------------|------|
| បុព្វកថា .....                                                   | i    |
| សេចក្តីបញ្ជាក់ .....                                             | vi   |
| អារម្ភកថា .....                                                  | vii  |
| សេចក្តីថ្លែងអំណរគុណ .....                                        | viii |
| មាតិកា .....                                                     | ix   |
| <b>មេរៀនទី១ សេចក្តីផ្តើមការអនុវត្តភាសា C</b> .....               | 1    |
| I. កម្មវិធី និង ភាសាកម្មវិធី.....                                | 1    |
| II. ហេតុអ្វីបានជាយើងរៀនភាសា C ? .....                            | 1    |
| III. របៀបបង្កើតកម្មវិធីលើភាសាកម្មវិធី C .....                    | 1    |
| IV. របៀបសរសេរកម្មវិធីលើភាសាកម្មវិធី C .....                      | 2    |
| V. គំរូកម្មវិធីដំបូង .....                                       | 3    |
| <b>មេរៀនទី២ ការប្រើប្រាស់ប្រភេទទិន្នន័យ</b> .....                | 4    |
| I. ប្រភេទទិន្នន័យ .....                                          | 4    |
| II. ការប្រើប្រាស់ប្រភេទ Integer .....                            | 4    |
| III. ការប្រើប្រាស់ប្រភេទ Floating-Point.....                     | 7    |
| IV. ការប្រើប្រាស់ប្រភេទ Void.....                                | 8    |
| <b>មេរៀនទី៣ ការប្រើអថេរ ចំនួនថេរនិងសញ្ញាប្រមាណវិធី</b> .....     | 10   |
| I. អថេរ.....                                                     | 10   |
| II. ការប្រកាសអថេរ .....                                          | 10   |
| III. ការផ្តល់តម្លៃអោយអថេរពេលប្រកាស .....                         | 10   |
| IV. ការប្រើសញ្ញាស្នើក្នុងការផ្តល់តម្លៃ .....                     | 12   |
| V. ចំនួនថេរ.....                                                 | 12   |
| VI. ចំនួនថេរ Literal .....                                       | 12   |
| VII. ចំនួនថេរ Define.....                                        | 14   |
| VIII. ចំនួនថេរ Memory .....                                      | 15   |
| IX. សញ្ញាប្រមាណវិធី .....                                        | 16   |
| <b>មេរៀនទី៤ ការប្រើប្រាស់បញ្ចប់បញ្ចូលនិងបញ្ចេញទិន្នន័យ</b> ..... | 26   |

|                                       |    |
|---------------------------------------|----|
| I. ការប្រើបញ្ជា Printf .....          | 26 |
| II. ការប្រើបញ្ជា Scanf .....          | 28 |
| III. ការប្រើបញ្ជា Textcolor .....     | 31 |
| IV. ការប្រើបញ្ជា TextBackground ..... | 33 |
| V. ការប្រើបញ្ជា Gotoxy .....          | 35 |
| VI. ការប្រើបញ្ជា Delay .....          | 37 |
| VII. ការប្រើបញ្ជា Clreol .....        | 39 |
| VIII. លំហាត់អនុវត្ត.....              | 41 |

## **មេរៀនទី៥ របៀបសរសេរកូដដោយប្រតិបត្តិតាមលក្ខខណ្ឌ .....**

|                                          |    |
|------------------------------------------|----|
| I. ការប្រើឃ្លាប្រតិបត្តិតាមលក្ខខណ្ឌ..... | 42 |
| II. ការប្រើឃ្លា if .....                 | 42 |
| III. ការប្រើឃ្លា if else .....           | 43 |
| IV. ការប្រើឃ្លា nested if .....          | 47 |
| V. ការប្រើឃ្លា switch .....              | 48 |
| VI. ការប្រើឃ្លា nested switch .....      | 51 |
| VII. ការប្រើសញ្ញា ? .....                | 52 |
| VIII. ការប្រើឃ្លា goto .....             | 54 |
| IX. ការប្រើឃ្លា exit .....               | 56 |
| X. ការប្រើឃ្លា tolower .....             | 58 |
| XI. ការប្រើឃ្លា toupper .....            | 60 |
| XII. ការប្រើឃ្លា kbhit.....              | 62 |
| XIII. ការប្រើ control loop .....         | 64 |
| XIV. ការប្រើ while.....                  | 65 |
| XV. ការប្រើ do while.....                | 66 |
| XVI. ការប្រើ for .....                   | 68 |
| XVII. ការប្រើ nested loop.....           | 70 |
| XVIII. ការប្រើ break .....               | 73 |
| XIX. ការប្រើ continue .....              | 75 |
| XX. ការប្រើ random.....                  | 76 |

|                                              |    |
|----------------------------------------------|----|
| XXI. ការប្រើ randomize .....                 | 77 |
| XXII. លំហាត់បង្ហាញពីការប្តូរលុយបីជំរើស ..... | 79 |

## **មេរៀនទី៦ ការបង្កើត Function .....**

|                                       |    |
|---------------------------------------|----|
| I. ការបង្កើត Function.....            | 83 |
| II. របៀបប្រកាស Function .....         | 84 |
| III. របៀបហៅ Function.....             | 84 |
| IV. ការប្រើ Parameter Function .....  | 85 |
| V. ការបង្កើត Recursion Function ..... | 87 |
| VI. ការបង្កើត Library File .....      | 89 |

## **មេរៀនទី៧ របៀបប្រើ Array មួយវិមាត្រ .....**

|                                                                 |     |
|-----------------------------------------------------------------|-----|
| I. ការប្រើ Array មួយវិមាត្រ.....                                | 91  |
| II. ការបង្កើត Array មួយវិមាត្រ .....                            | 91  |
| III. របៀបប្រើ Array មួយវិមាត្រ .....                            | 91  |
| IV. ការប្រើ Array មួយវិមាត្រជាមួយ Function .....                | 94  |
| V. ការប្រើ return Array ពី Function .....                       | 96  |
| VI. ការប្រើ Search .....                                        | 98  |
| VII. ការប្រើ Sort.....                                          | 100 |
| VIII. ការប្រើ Delete .....                                      | 104 |
| IX. ការប្រើ Edit .....                                          | 107 |
| X. លំហាត់ការគ្រប់គ្រងព័ត៌មានសិស្សចុះឈ្មោះចូលរៀនសាលារៀនមួយ ..... | 109 |

## **មេរៀនទី៨ របៀបប្រើ Array ច្រើនវិមាត្រ.....**

|                                          |     |
|------------------------------------------|-----|
| I. ការប្រើ Array ច្រើនវិមាត្រ .....      | 120 |
| II. ការបង្កើត Array ពីវិមាត្រ .....      | 120 |
| III. ការប្រើ Array ពីវិមាត្រ .....       | 121 |
| IV. ការប្រើ Array ពីវិមាត្រជាអក្សរ ..... | 122 |
| V. ការប្រើ Search .....                  | 123 |
| VI. ការប្រើ Sort.....                    | 124 |
| VII. ការប្រើ Delete .....                | 126 |
| VIII. ការប្រើ Edit .....                 | 128 |

|                                                                            |     |
|----------------------------------------------------------------------------|-----|
| IX. លំហាត់ការគ្រប់គ្រងព័ត៌មានសិស្សចុះឈ្មោះចូលរៀនសាលារៀនមួយ (Update) .....  | 130 |
| <b>មេរៀនទី៩ របៀបប្រើ String</b> .....                                      | 142 |
| I. និយមន័យ String .....                                                    | 142 |
| II. របៀបប្រើ String.....                                                   | 142 |
| III. ការប្រើឃ្លាបញ្ចាមួយ String.....                                       | 143 |
| IV. ការប្រើប្រតិបត្តិលក្ខខណ្ឌ String .....                                 | 145 |
| V. ការបំប្លែង String ទៅជាលេខ .....                                         | 147 |
| VI. ការបំប្លែងលេខទៅជា String .....                                         | 148 |
| VII. លំហាត់ការគ្រប់គ្រងព័ត៌មានសិស្សចុះឈ្មោះចូលរៀនសាលារៀនមួយ (Update) ..... | 150 |
| <b>មេរៀនទី១០ ការប្រើប្រាស់ Pointer</b> .....                               | 163 |
| I. និយមន័យ Pointer .....                                                   | 163 |
| II. របៀបប្រើ Pointer.....                                                  | 163 |
| III. របៀបប្រើ Pointer arithmetic .....                                     | 164 |
| IV. របៀបប្រើ Pointer និង Array.....                                        | 166 |
| V. របៀបប្រើ Pointer និង Pointer.....                                       | 169 |
| VI. របៀបប្រើ Pointer និង Function .....                                    | 170 |
| VII. របៀបប្រើ Pointer និង Memory Management.....                           | 174 |
| <b>មេរៀនទី១១ ការប្រើប្រាស់ Structure</b> .....                             | 178 |
| I. និយមន័យ Structure .....                                                 | 178 |
| II. ការបង្កើត Structure .....                                              | 178 |
| III. របៀបប្រើ Structure .....                                              | 179 |
| IV. របៀបប្រើ Structure និង Array .....                                     | 180 |
| V. របៀបប្រើ Structure និង Pointer .....                                    | 182 |
| VI. របៀបប្រើ Structure និង Function.....                                   | 183 |
| VII. ការប្រើ Search .....                                                  | 185 |
| VIII. ការប្រើ Sort.....                                                    | 188 |
| IX. ការប្រើ Delete .....                                                   | 190 |
| X. ការប្រើ Edit .....                                                      | 193 |
| XI. លំហាត់ការគ្រប់គ្រងផ្នែកលក់ទំនិញក្នុងហាងមួយ .....                       | 196 |

|                                                                  |            |
|------------------------------------------------------------------|------------|
| <b>មេរៀនទី១២ របៀបបង្កើតនិងប្រើប្រាស់ Text File .....</b>         | <b>213</b> |
| I. និយមន័យ Text File .....                                       | 213        |
| II. ការប្រើបញ្ជា fopen .....                                     | 213        |
| III. ការប្រើបញ្ជា fclose .....                                   | 214        |
| IV. ការប្រើបញ្ជា រក្សាទុកទិន្នន័យ .....                          | 214        |
| V. ការប្រើបញ្ជា អានទិន្នន័យ .....                                | 215        |
| VI. ការប្រើបញ្ជា rename .....                                    | 217        |
| VII. ការប្រើបញ្ជា remove.....                                    | 217        |
| VIII. ការប្រើ Search .....                                       | 218        |
| IX. ការប្រើ Sort.....                                            | 220        |
| X. ការប្រើ Delete .....                                          | 223        |
| XI. ការប្រើ Edit .....                                           | 227        |
| XII. លំហាត់ការគ្រប់គ្រងផ្នែកលក់ទំនិញក្នុងហាងមួយ ( Update ).....  | 231        |
| <b>មេរៀនទី១៣ របៀបបង្កើតនិងប្រើប្រាស់ Binary File .....</b>       | <b>250</b> |
| I. និយមន័យ Binary File.....                                      | 250        |
| II. ការប្រើបញ្ជា fopen .....                                     | 250        |
| III. ការប្រើបញ្ជា fclose .....                                   | 251        |
| IV. ការប្រើបញ្ជា fwrite.....                                     | 251        |
| V. ការប្រើបញ្ជា fread .....                                      | 252        |
| VI. ការប្រើបញ្ជា ftell.....                                      | 253        |
| VII. ការប្រើបញ្ជា rewind.....                                    | 254        |
| VIII. ការប្រើបញ្ជា fseek.....                                    | 255        |
| IX. ការប្រើ Search .....                                         | 257        |
| X. ការប្រើ Sort.....                                             | 259        |
| XI. ការប្រើ Delete .....                                         | 263        |
| XII. ការប្រើ Update .....                                        | 268        |
| XIII. លំហាត់ការគ្រប់គ្រងផ្នែកលក់ទំនិញក្នុងហាងមួយ ( Update )..... | 273        |
| <b>ឯកសារពិគ្រោះ: .....</b>                                       | <b>297</b> |



I. កម្មវិធី និង ភាសាកម្មវិធី

ភាសាកំណើតរបស់កុំព្យូទ័រគឺប្រព័ន្ធគោលពីរ មួយនិងសូន្យហើយការណែនាំនិងទិន្នន័យទាំងអស់ត្រូវតែបានផ្តល់ឱ្យវានៅក្នុងទម្រង់នេះ។ លេខកូដគោលពីរដើមត្រូវបានគេហៅថាភាសាម៉ាស៊ីន។ ដូច្នេះកម្មវិធីដែលយើងប្រើប្រាស់ក្នុងម៉ាស៊ីនត្រូវបានកំណត់ឡើងមកតាមរយៈភាសាម៉ាស៊ីនសំរាប់ប្រើប្រាស់ម៉ាស៊ីនអោយធ្វើអ្វីមួយដែលយើងចង់បាន។

II. ហេតុអ្វីបានជាយើងរៀនភាសា C ?

ភាសាសរសេរកម្មវិធី C គឺជាភាសាចាំបាច់សម្រាប់សិស្សានុសិស្ស និងអ្នកជំនាញដែលធ្វើការដើម្បីក្លាយជាវិស្វករ Software ដ៏អស្ចារ្យ ជាពិសេសនៅពេលដែលពួកគេកំពុងធ្វើការនៅក្នុងកម្មវិធីអភិវឌ្ឍន៍។ ខាងក្រោមជាបញ្ជីអត្ថប្រយោជន៍សំខាន់ៗមួយចំនួននៃការរៀន C Programming

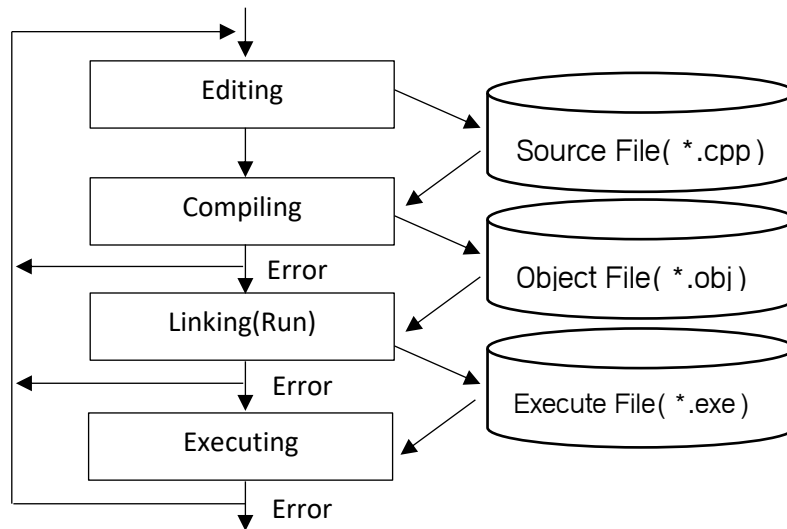
- ងាយស្រួលរៀន
- ភាសាដែលមានរចនាសម្ព័ន្ធ
- របៀបបង្កើតកម្មវិធីមិនស្មុគស្មាញ
- កុំព្យូទ័រមិនត្រូវការកំរិតខ្ពស់

III. របៀបបង្កើតកម្មវិធីលើភាសាកម្មវិធី C

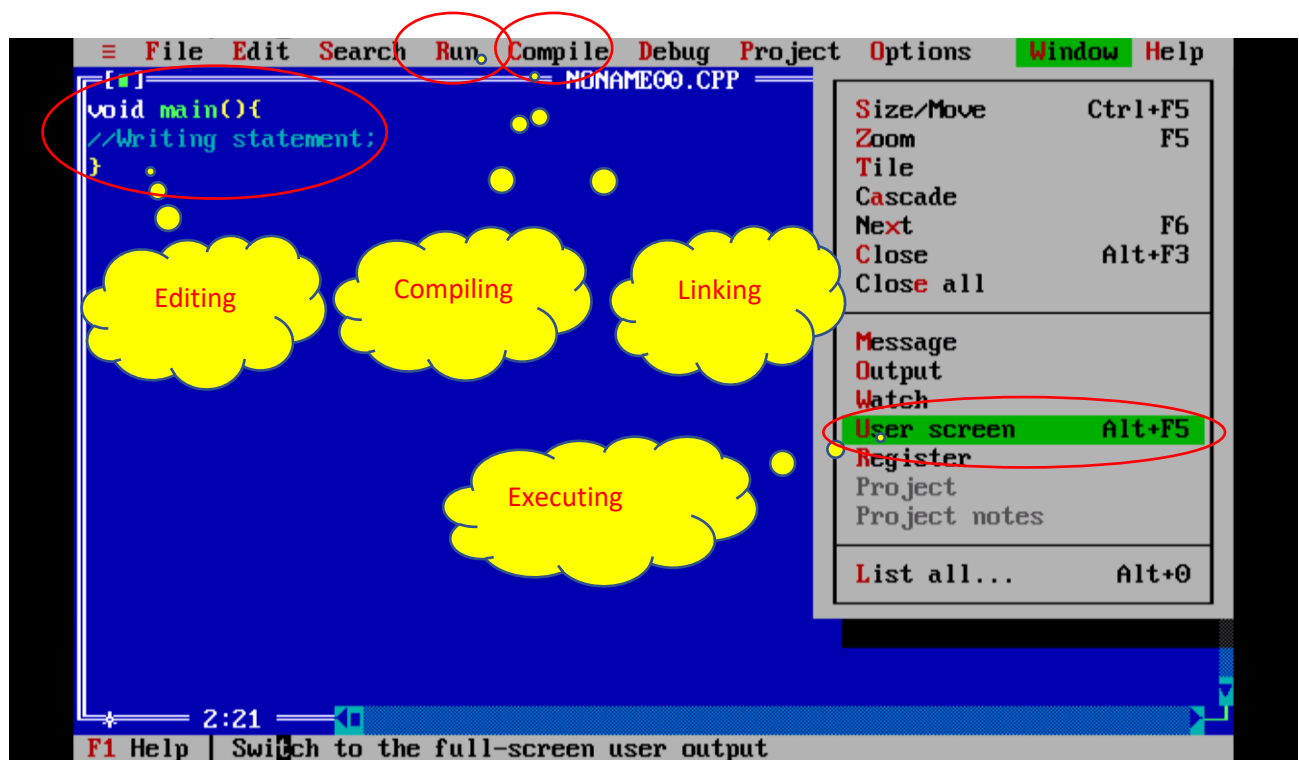
ដើម្បីបង្កើតចេញជាកម្មវិធីមួយពីភាសាC យើងមានចំណុចសំខាន់ៗដូចខាងក្រោម

- Editing: យើងត្រូវចុះសរសេរ Source code
- Compiling: យើងត្រូវចុះចុច Compile ( alt + F9 )
- Liking( Run ):យើងត្រូវចុះចុច Run ( Ctrl + F9 )
- Executing: យើងត្រូវចុះតាមដានមើលលទ្ធផល ( alt + F5 )

ខាងក្រោមជាការបង្ហាញអំពីដំណើរនៃប្រតិបត្តិការរបស់ដំហ៊ានបង្កើតកម្មវិធីលើភាសាកម្មវិធី C ត្រូវបានគូចេញជា Flow chat



ខាងក្រោមជាការបង្ហាញអំពីស្ថានភាពទូទៅដែលមានប្រើក្នុងកម្មវិធី C



#### IV. របៀបសរសេរកម្មវិធីលើភាសាកម្មវិធី C

ក្នុងផ្ទាំង Window សរសេរ code ចែកចេញជាប្លុកផ្នែកដូចខាងក្រោម

1. Preprocessor Directive
2. Global Declaration

```
3. void main( ){
    // statement
}
```

```
4. void function( ){
    // statement
}
```

ឧទាហរណ៍:

```
#include<stdio.h>
```

```
void show( );
```

```
void main( ){    show( );
}
```

```
void show( ) { printf( "Welcome to RUPP" ); }
```

#### V. គំរូកម្មវិធីដំបូង

ខាងក្រោមជាការសរសេរកម្មវិធីមួយដោយម៉ាស៊ីនបង្ហាញអត្ថន័យ "Hello! RUPP" នៅលើScreen Computer

```
#include<stdio.h>
```

```
void main( ){
    printf( "Hello! RUPP" );
}
```

**ខាងក្រោមជាលទ្ធផលនៃកម្មវិធី**

Hello! RUPP

## មេរៀនទី២

## ការប្រើប្រាស់ប្រភេទទិន្នន័យ

### I. ប្រភេទទិន្នន័យ

ប្រភេទទិន្នន័យនៅក្នុងភាសា C សំដៅលើការកំណត់ដែនតម្លៃអោយអថេរ ឬ ចំនួនថេរដែលមានប្រភេទផ្សេងៗគ្នា។ នៅក្នុងភាសា C អាចបែងចែកប្រភេទទិន្នន័យមានដូចខាងក្រោម

| ល | ប្រភេទ និង ការពិពណ៌នា                                                                                                                                 |
|---|-------------------------------------------------------------------------------------------------------------------------------------------------------|
| 1 | <b>Basic Types</b><br>ជាប្រភេទ arithmetic ដែលមានពីរប្រភេទ : (a) integer types និង (b) floating-point types.                                           |
| 2 | <b>Enumerated types</b><br>វាក៏ជាប្រភេទ arithmetic ដែរត្រូវបានប្រើក្នុងការផ្តល់តម្លៃជាចំនួនគត់ជាក់លាក់អោយទៅ variables ទាំងអស់ដែលមានប្រើក្នុងកម្មវិធី។ |
| 3 | <b>The type void</b><br>ជាប្រភេទ void គឺជាប្រភេទដែលបញ្ជាក់ថាគ្មានតម្លៃទេ។                                                                             |
| 4 | <b>Derived types</b><br>ជាប្រភេទដូចជា (a) Pointer types, (b) Array types, (c) Structure types, (d) Union types and (e) Function types.                |

### II. ការប្រើប្រាស់ប្រភេទ Integer

តារាងខាងក្រោមផ្តល់នូវព័ត៌មានលម្អិតនៃប្រភេទចំនួនគត់ស្តង់ដារជាមួយនឹងទំហំផ្ទុក និងជួរតម្លៃរបស់ពួកគេ –

| Type          | Storage size | Value range             |
|---------------|--------------|-------------------------|
| char          | 1 byte       | -128 to 127 or 0 to 255 |
| unsigned char | 1 byte       | 0 to 255                |

|                |                                     |                                                             |
|----------------|-------------------------------------|-------------------------------------------------------------|
| signed char    | 1 byte                              | -128 to 127                                                 |
| int            | 2 or 4 bytes                        | -32,768 to 32,767 or -<br>2,147,483,648 to<br>2,147,483,647 |
| unsigned int   | 2 or 4 bytes                        | 0 to 65,535 or 0 to<br>4,294,967,295                        |
| short          | 2 bytes                             | -32,768 to 32,767                                           |
| unsigned short | 2 bytes                             | 0 to 65,535                                                 |
| long           | 8 bytes or ( 4bytes for 32 bit OS ) | -9223372036854775808 to<br>9223372036854775807              |
| unsigned long  | 8 bytes                             | 0 to<br>18446744073709551615                                |

ខាងក្រោមជាលំហាត់អនុវត្តន៍បង្ហាញអំពីទំហំទិន្នន័យជាក់លាក់របស់ប្រភេទទិន្នន័យនីមួយៗដែលយើងមានប្រើក្នុងតារាងខាងលើ។

```
#include <stdio.h>
```

```
#include <stdlib.h>
```

```
#include <limits.h>
```

```
#include <float.h>
```

```
int main( ) {
```

```
    printf( "CHAR_BIT   : %d\n", CHAR_BIT );
```

```

printf("CHAR_MAX   : %d\n", CHAR_MAX);

printf("CHAR_MIN   : %d\n", CHAR_MIN);

printf("INT_MAX     : %d\n", INT_MAX);

printf("INT_MIN     : %d\n", INT_MIN);

printf("LONG_MAX    : %ld\n", (long) LONG_MAX);

printf("LONG_MIN    : %ld\n", (long) LONG_MIN);

printf("SCHAR_MAX   : %d\n", SCHAR_MAX);

printf("SCHAR_MIN   : %d\n", SCHAR_MIN);

printf("SHRT_MAX    : %d\n", SHRT_MAX);

printf("SHRT_MIN    : %d\n", SHRT_MIN);

printf("UCHAR_MAX   : %d\n", UCHAR_MAX);

printf("UINT_MAX     : %u\n", (unsigned int) UINT_MAX);

printf("ULONG_MAX    : %lu\n", (unsigned long) ULONG_MAX);

printf("USHRT_MAX    : %d\n", (unsigned short) USHRT_MAX);

return 0;

}

```

### **ខាងក្រោមជារង្វង់ដែលបានពីកម្មវិធីខាងលើ**

```

CHAR_BIT   : 8

CHAR_MAX   : 127

CHAR_MIN   : -128

INT_MAX    : 2147483647

INT_MIN    : -2147483648

```

LONG\_MAX : 9223372036854775807

LONG\_MIN : -9223372036854775808

SCHAR\_MAX : 127

SCHAR\_MIN : -128

SHRT\_MAX : 32767

SHRT\_MIN : -32768

UCHAR\_MAX : 255

UINT\_MAX : 4294967295

ULONG\_MAX : 18446744073709551615

USHRT\_MAX : 65535

### III. ការប្រើប្រាស់ប្រភេទ Floating-Point

តារាងខាងក្រោមផ្តល់នូវព័ត៌មានលម្អិតនៃប្រភេទ Floating-Point ជាមួយនឹងទំហំផ្ទុក និងជួរតម្លៃរបស់ពួកគេ –

| Type        | Storage size | Value range            | Precision         |
|-------------|--------------|------------------------|-------------------|
| float       | 4 byte       | 1.2E-38 to 3.4E+38     | 6 decimal places  |
| double      | 8 byte       | 2.3E-308 to 1.7E+308   | 15 decimal places |
| long double | 10 byte      | 3.4E-4932 to 1.1E+4932 | 19 decimal places |

ខាងក្រោមជាលំហាត់អនុវត្តន៍បង្ហាញអំពីទំហំទិន្នន័យជាក់លាក់របស់ប្រភេទទិន្នន័យនីមួយៗដែលយើងមានប្រើក្នុងតារាងខាងលើ។

```
#include <stdio.h>
```

```
#include <stdlib.h>
```

```
#include <limits.h>
```

```
#include <float.h>
```

```

int main( ) {
    printf( "Storage size for float : %d \n", sizeof( float ) );
    printf( "FLT_MAX    : %g\n", ( float ) FLT_MAX );
    printf( "FLT_MIN    : %g\n", ( float ) FLT_MIN );
    printf( "-FLT_MAX    : %g\n", ( float ) -FLT_MAX );
    printf( "-FLT_MIN    : %g\n", ( float ) -FLT_MIN );
    printf( "DBL_MAX    : %g\n", ( double ) DBL_MAX );
    printf( "DBL_MIN    : %g\n", ( double ) DBL_MIN );
    printf( "-DBL_MAX    : %g\n", ( double ) -DBL_MAX );
    printf( "Precision value: %d\n", FLT_DIG );

    return 0;
}

```

### **ខាងក្រោមជាលទ្ធផលដែលបានពីកម្មវិធីខាងលើ**

Storage size for float : 4

FLT\_MAX : 3.40282e+38

FLT\_MIN : 1.17549e-38

-FLT\_MAX : -3.40282e+38

-FLT\_MIN : -1.17549e-38

DBL\_MAX : 1.79769e+308

DBL\_MIN : 2.22507e-308

-DBL\_MAX : -1.79769e+308

Precision value: 6

#### **IV. ការប្រើប្រាស់ប្រភេទ Void**

ប្រភេទ void បញ្ជាក់ថាគ្មានតម្លៃទេ។ វាត្រូវបានប្រើក្នុងស្ថានភាពបីប្រភេទ –



| Sr.No. | ប្រភេទ និង ការពិពណ៌នា                                                                                                                                                                                                                   |
|--------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 1      | <p><b>Function returns as void</b></p> <p>មាន Function ប្រភេទផ្សេងៗគ្នានៅក្នុងភាសា C ដែលអាចមាន return រឺ មិនមាន return។ សំរាប់ Function មិនមាន return ត្រូវមាន Function type void, example void exit( int status );</p>                 |
| 2      | <p><b>Function arguments as void</b></p> <p>មាន Function ប្រភេទផ្សេងៗគ្នានៅក្នុងភាសា C ដែលអាចមាន Parameter រឺ មិនមាន Parameter ។ សំរាប់ Function មិនមាន Parameter ត្រូវគេហៅថា Function argument as void , example int rand( void );</p> |
| 3      | <p><b>Pointers to void</b></p> <p>សំរាប់ Pointer to type void * គំណាងឱ្យអាសយដ្ឋានរបស់ Object ណាមួយដែរមិនមែនជាប្រភេទរបស់វា។ ឧទាហរណ៍ void *malloc( size_t size ); ក៏វា return a pointer to void អាចប្រើបានគ្រប់ប្រភេទទិន្នន័យ។</p>        |

# មេរៀនទី៣

# ការប្រើអថេរ ចំនួនថេរនិងសញ្ញាប្រមាណវិធី

## I. អថេរ

អថេរ (variable) គឺជាឈ្មោះត្រូវបានបង្កើតឡើងក្នុងដំបន់ Memory គោលបំណងរក្សាទុកទិន្នន័យទាំងឡាយ ដែលមានប្រើប្រាស់ក្នុងកម្មវិធី។ អថេរនីមួយៗប្រើប្រាស់ត្រូវមានប្រភេទទិន្នន័យកំណត់ជាក់លាក់។

ការកំណត់ឈ្មោះរបស់អថេរហាមបំរាមចំនួន៤ដូចខាងក្រោម

- ហាមសរសេរដកឃ្លា(បើចាំបាច់អាចប្រើសញ្ញា “\_” Underscore ចំនួនបាន )
- ហាមនាំមុខដោយសញ្ញា
- ហាមចូលរួមដោយសញ្ញាគណនា
- ហាមដូច Keyword ( ជាពាក្យដែលមានន័យរួចហើយសរសេរគ្រប់គួរអក្សរនៃពាក្យទៅចេញពត៌ស )

## II. ការប្រកាសអថេរ

ខាងក្រោមជាទំរង់ក្នុងការប្រកាសអថេរប្រើក្នុងកម្មវិធី

Syntax:            datatype identifier;

ឧទាហរណ៍

```
int    i, j, k;
char   c, ch;
char   st[20];
float  f, salary;
```

នៅបន្ទាត់     int i, j, k; មានន័យថា i, j និង k មានប្រភេទទិន្នន័យដូចគ្នាជាប្រភេទចំនួនគត់

នៅបន្ទាត់     char c,ch; មានន័យថា c និង ch មានប្រភេទទិន្នន័យដូចគ្នាជា អក្សរមួយ

នៅបន្ទាត់     char st[20]; មានន័យថា st មានប្រភេទទិន្នន័យជា អក្សរប្រើន

នៅបន្ទាត់     float f,salary; មានន័យថា f និង salary មានប្រភេទទិន្នន័យដូចគ្នាជាចំនួនទសភាគ

## III. ការផ្តល់តម្លៃអោយអថេរពេលប្រកាស

ការប្រកាសអថេរយើងអាចអោយតម្លៃនៅពេលប្រកាសវាបាន។ខាងក្រោមជាទំរង់ទូទៅ

Syntax: datatype identifier = value;

ឧទាហរណ៍

```
int    i = 0, j = 3, k = 5;
char   c='k', ch='$';
char   st[20]="Hello!";
float  f=0.0 , salary= 7.5 ;
```

## ខាងក្រោមជាលំហាត់អនុវត្តក្នុងការប្រកាសអថេរ

```
#include <stdio.h>

// Variable declaration:

int main ( ) {

    /* variable definition: */

    int a, b;

    int c;

    float f;

    /* actual initialization */

    a = 10;

    b = 20;

    c = a + b;

    printf("value of c : %d \n", c);

    f = 70.0/3.0;

    printf("value of f : %f \n", f);

    return 0;

}
```

## ខាងក្រោមជាលទ្ធផលលំហាត់ខាងលើ

```
value of c : 30
value of f : 23.333334
```

#### IV. ការប្រើសញ្ញាស្មើក្នុងការផ្តល់តម្លៃ

សញ្ញាស្មើជាបញ្ហាសំរាប់ប្រើក្នុងការផ្តល់តម្លៃ។ តម្លៃអាចមាននៅខាងធ្វេងនិងខាងស្តាំនៃសញ្ញាស្មើតែមានន័យផ្សេងគ្នា។ ខាងធ្វេងនៃសញ្ញាស្មើត្រូវតែជាអ្នកទទួលតម្លៃអាចជា Variable ឬ Constant ហើយខាងស្តាំនៃសញ្ញាស្មើត្រូវតែជាអ្នកផ្តល់តម្លៃ។

```
ឧទាហរណ៍      int      a = 20;      // ត្រឹមត្រូវ
                  10 = 20;      // មិនត្រឹមត្រូវ Error
```

#### V. ចំនួនថេរ

ចំនួនថេរសំដៅលើតម្លៃថេរដែលកម្មវិធីអាចមិនត្រូវការផ្លាស់ប្តូរកំឡុងពេលប្រតិបត្តិរបស់វា។ តម្លៃថេរទាំងនេះត្រូវបានគេហៅវាបានបីរបៀបដូចខាងក្រោម។

- ចំនួនថេរ Literal
- ចំនួនថេរ Define
- ចំនួនថេរ Memory

#### VI. ចំនួនថេរ Literal

ចំនួនថេរ Literal អាចជាទិន្នន័យណាមួយដូចជាតម្លៃថេរចំនួនគត់ តម្លៃថេរចំនួនទសភាគ តម្លៃថេរជាតួអក្សរមួយ ឬ អក្សរច្រើនត្រូវបានប្រើនៅក្នុងឃ្លាបញ្ជាហើយមានន័យថាជាតម្លៃដែលមិនចេះប្រែប្រួល។

##### 1. តម្លៃថេរចំនួនគត់

**ខាងក្រោមជាឧទាហរណ៍តម្លៃថេរចំនួនគត់ដែលមានប្រើក្នុងការសរសេរ**

```
85      /* decimal */
0213    /* octal */
0x4b    /* hexadecimal */
30      /* int */
30u     /* unsigned int */
30l     /* long */
30ul    /* unsigned long */
```

Ex:

```
#include<stdio.h>
#include<conio.h>
```

```

int a = 5;
int b = 10;
void main( ){
    clrscr( );
    printf( "value a is: %d\n ", a );
    printf( "value b is: %d\n",b );
    getch( );
}

```

## 2. តម្លៃថេរចំនួនទសភាគ

ខាងក្រោមជាឧទាហរណ៍តម្លៃថេរចំនួនទសភាគដែលមានប្រើក្នុងការសរសេរ

```

3.14159    /* Legal */
314159E-5L  /* Legal */
510E       /* Illegal: incomplete exponent */
210f       /* Illegal: no decimal or exponent */
.e55       /* Illegal: missing integer or fraction */

```

Ex:

```

#include<stdio.h>
#include<conio.h>
float a = 3.5;
float b = 2.3;
void main( ){
    clrscr( );
    printf( "value a is: %f\n ", a );
    printf( "value b is: %f\n",b );
    getch( );
}

```

## 3. តម្លៃថេរជាអក្សរមួយ

តម្លៃថេរជាអក្សរមួយត្រូវបានគេប្រើក្នុងការសរសេរដោយដាក់នៅចន្លោះសញ្ញា Single quote ឧ: 'x' ហើយអាចរក្សាទុកក្នុងអថេរដែលមានប្រភេទទិន្នន័យជា char ។ ខាងក្រោមជាឧទាហរណ៍នៃការប្រើ

```

char x = '?';

```

```
char sex='F';
```

Ex:

```
#include<stdio.h>
#include<conio.h>
char sex = 'F';
char x = '?' ;
void main( ){
    clrscr( );
    printf( "value Sex is: %c\n ", sex );
    printf( "value x is : %c\n", x );
    getch( );
}
```

#### 4. តម្លៃថេរជាអក្សរច្រើន

តម្លៃថេរជាអក្សរច្រើនត្រូវបានគេប្រើក្នុងការសរសេរដោយដាក់នៅចន្លោះសញ្ញា Double quote ឧ:  
"Hello!" ហើយអាចរក្សាទុកក្នុងអថេរដែលមានប្រភេទទិន្នន័យជា char ។ ខាងក្រោមជាឧទាហរណ៍

```
char pro_name[20]="cocacola";
```

```
char person_name[30]="Bopha kok";
```

Ex:

```
#include<stdio.h>
#include<conio.h>
char pro_name[20]="Cocacola";
char person_name[30]="Bopha kok";
void main( ){
    clrscr( );
    printf( "Customer name is %s\n ", person_name );
    printf( "Order product name is %s\n", pro_name );
    getch( );
}
```

## VII. ចំនួនថេរ Define

ចំនួនថេរ Define អាចជាទិន្នន័យណាមួយដូចជាតម្លៃថេរចំនួនគត់ តម្លៃថេរចំនួនទសភាគ តម្លៃថេរជាតួអក្សរ មួយ ឬ អក្សរច្រើនត្រូវបានប្រកាសដោយពាក្យ #define នៅផ្នែក Preprocessor ហើយមានន័យថាជាតម្លៃ ដែលមិនចេះប្រែប្រួលមានឈ្មោះ។ ខាងក្រោមជាឧទាហរណ៍

Syntax:

```
#define identifier value
```

Ex:

```
#include <stdio.h>
```

```
#define LENGTH 10
```

```
#define WIDTH 5
```

```
#define NEWLINE '\n'
```

```
int main( ) {
```

```
    int area;
```

```
    area = LENGTH * WIDTH;
```

```
    printf("value of area : %d", area );
```

```
    printf("%c", NEWLINE);
```

```
    return 0;
```

```
}
```

**ខាងក្រោមជាលទ្ធផលរបស់កម្មវិធីខាងលើ**

value of area : 50

## VIII. ចំនួនថេរ Memory

ចំនួនថេរ Memory អាចជាទិន្នន័យណាមួយដូចជាតម្លៃថេរចំនួនគត់ តម្លៃថេរចំនួនទសភាគ តម្លៃថេរជាតួអក្សរ មួយ ឬ អក្សរច្រើនត្រូវបានប្រកាសដោយពាក្យ const នៅពីខាងមុខការប្រកាសអថេរណាដែលបាននិយាយ តម្លៃហើយមានន័យថាអថេរនោះមិនចេះប្រែប្រួលតម្លៃទេ។ ខាងក្រោមជាឧទាហរណ៍

```
const datatype identifier = value;
```

Ex:

```
#include <stdio.h>

int main( ) {

    const int LENGTH = 10;

    const int WIDTH = 5;

    const char NEWLINE = '\n';

    int area;

    area = LENGTH * WIDTH;

    printf("value of area : %d", area );

    printf("%c", NEWLINE );

    return 0;

}
```

**ខាងក្រោមជានិមិត្តរូបរបស់កម្មវិធីខាងលើ**

value of area : 50

## IX. សញ្ញាប្រមាណវិធី

សញ្ញាប្រមាណវិធីគឺជានិមិត្តសញ្ញាដែលប្រាប់ Compiler ឱ្យអនុវត្តមុខងារគណិតវិទ្យា ឬតក្កវិជ្ជាជាក់លាក់។ ភាសា C សម្បូរទៅដោយសញ្ញាប្រមាណវិធីដែលភ្ជាប់មកជាមួយ ហើយផ្តល់នូវប្រភេទប្រតិបត្តិការខាងក្រោម

- Arithmetic Operators
  - Relational Operators
  - Logical Operators
  - Bitwise Operators
  - Assignment Operators
  - Misc Operators
1. Arithmetic Operators



Arithmetic ជាសញ្ញាដែលប្រើក្នុងការធ្វើប្រមាណវិធីគណនាតម្លៃជាប្រភេទលេខ។ ខាងក្រោមជាតារាងបង្ហាញពីនិមិត្តសញ្ញា Arithmetic ។

| Operator | Description                                                  | Example<br>(A=10,B=20) |
|----------|--------------------------------------------------------------|------------------------|
| +        | Adds two operands.                                           | A + B = 30             |
| -        | Subtracts second operand from the first.                     | A - B = -10            |
| *        | Multiplies both operands.                                    | A * B = 200            |
| /        | Divides numerator by de-numerator.                           | B / A = 2              |
| %        | Modulus Operator and remainder of after an integer division. | B % A = 0              |
| ++       | Increment operator increases the integer value by one.       | A++ = 11               |
| --       | Decrement operator decreases the integer value by one.       | A-- = 9                |

### ខាងក្រោមជាលំហាត់អំពីការប្រើនិមិត្តសញ្ញា Arithmetic

Ex:        `#include<stdio.h>`

`#include<conio.h>`

`int A=10;`

`int B=20;`

`void main( ){`

`clrscr( );`

`printf(“ %d + %d = %d\n”, A , B, A+B);`

```

printf( " %d - %d = %d\n", A , B, A-B);

printf( " %d * %d = %d\n", A , B, A*B);

printf( " %d / %d = %d\n", B , A, B/A);

printf( " %d %% %d = %d\n", B , A, B%A);

printf( " value A++ = %d \n", A ++ );

printf( " value A-- = %d\n", A-- );

getch( );

}

```

### ខាងក្រោមជាលទ្ធផលរបស់កម្មវិធីខាងលើ

10 + 20 = 30

10 - 20 = -10

10 \* 20 = 200

20 / 10 = 2

20 % 10 = 0

Value A++ = 11

Value A-- = 9

## 2. Relational Operators

Relational ជាសញ្ញាដែលប្រើក្នុងការធ្វើប្រមាណវិធីប្រៀបធៀបតម្លៃជាប្រភេទលេខ។ ខាងក្រោមជាតារាងបង្ហាញពីនិមិត្តសញ្ញា Relational ។

| Operator | ការពិពណ៌នា | Example( A=1,B=2 ) |
|----------|------------|--------------------|
|----------|------------|--------------------|

|    |                                                                                                                                      |                         |
|----|--------------------------------------------------------------------------------------------------------------------------------------|-------------------------|
| == | Checks if the values of two operands are equal or not. If yes, then the condition becomes true.                                      | ( A == B ) is not true. |
| != | Checks if the values of two operands are equal or not. If the values are not equal, then the condition becomes true.                 | ( A != B ) is true.     |
| >  | Checks if the value of left operand is greater than the value of right operand. If yes, then the condition becomes true.             | ( A > B ) is not true.  |
| <  | Checks if the value of left operand is less than the value of right operand. If yes, then the condition becomes true.                | ( A < B ) is true.      |
| >= | Checks if the value of left operand is greater than or equal to the value of right operand. If yes, then the condition becomes true. | ( A >= B ) is not true. |
| <= | Checks if the value of left operand is less than or equal to the value of right operand. If yes, then the condition becomes true.    | ( A <= B ) is true.     |

### **ខាងក្រោមជាលំហាត់អំពីការប្រើនិមិត្តសញ្ញា Relational ( true = 1, false = 0 )**

Ex:

```
#include<stdio.h>

#include<conio.h>

int A=1;

int B=2;

void main( ){
```

```

clrscr( );

printf( " if %d == %d is %d\n", A , B, A==B);

printf( " if %d != %d is %d\n", A , B, A!=B);

printf( " if %d > %d is %d\n", A , B, A<B);

printf( " if %d < %d is %d\n", A , B, A<B);

printf( " if %d >= %d is %d\n", A , B, A>=B);

printf( " if %d <= %d is %d", A, B, A<=B);

getch( );

}

```

### ខាងក្រោមជាលទ្ធផលរបស់កម្មវិធីខាងលើ

if 1 == 2 is 0

if 1 != 2 is 1

if 1 > 2 is 0

if 1 < 2 is 1

if 1 >= 2 is 0

if 1 <= 2 is 1

### 3. Logical Operators

Logical ជាសញ្ញាដែលប្រើក្នុងការធ្វើប្រមាណវិធីជាផ្ទាល់ភ្ជាប់លក្ខខណ្ឌច្រើន។ ខាងក្រោមជាតារាងបង្ហាញពីនិមិត្តសញ្ញា Logical ។

| Operator | ការពិពណ៌នា                                                                                       | Example(A=1,B=0)   |
|----------|--------------------------------------------------------------------------------------------------|--------------------|
| &&       | Called Logical AND operator. If both the operands are non-zero, then the condition becomes true. | (A && B) is false. |

|   |                                                                                                                                                            |                    |
|---|------------------------------------------------------------------------------------------------------------------------------------------------------------|--------------------|
|   | Called Logical OR Operator. If any of the two operands is non-zero, then the condition becomes true.                                                       | (A    B) is true.  |
| ! | Called Logical NOT Operator. It is used to reverse the logical state of its operand. If a condition is true, then Logical NOT operator will make it false. | !(A && B) is true. |

## ខាងក្រោមជាលំហាត់អំពីការប្រើនិងមតិសញ្ញា Logical ( true = 1, false = 0 )

Ex:

```
#include<stdio.h>

#include<conio.h>

int A=1;

int B=0;

void main( ){

    clrscr( );

    printf(" if %d && %d is %d\n ", A , B, A&&B);

    printf(" if %d || %d is %d\n", A , B, A||B);

    printf(" if !(%d && %d) is %d\n", A , B, !(A&&B));

    getch( );

}
```

## ខាងក្រោមជាលទ្ធផលរបស់កម្មវិធីខាងលើ

if 1 && 0 is 0

if 1 || 0 is 1

if !(1&&0) is 1

## 4. Bitwise Operators

Bitwise ជាសញ្ញាដែលប្រើក្នុងការធ្វើប្រមាណវិធីជា Bit by Bit ។ ខាងក្រោមជាតារាងបង្ហាញពីនិមិត្តសញ្ញា Bitwise ។

| p | q | p & q | p   q | p ^ q |
|---|---|-------|-------|-------|
| 0 | 0 | 0     | 0     | 0     |
| 0 | 1 | 0     | 1     | 1     |
| 1 | 1 | 1     | 1     | 0     |
| 1 | 0 | 0     | 1     | 1     |

### ខាងក្រោមជាលំហាត់អំពីការប្រើនិមិត្តសញ្ញា Bitwise

Ex:

```
#include<stdio.h>

#include<conio.h>

int p;

int q;

void main( ){

    clrscr( );

    p=0; q=0;

    printf( " if p=%d , q=%d is p&q=%d, p|q=%d, p^q=%d\n ",p,q,p&q,p|q,p^q );

    p=0; q=1;

    printf( " if p=%d , q=%d is p&q=%d, p|q=%d, p^q=%d\n ",p,q,p&q,p|q,p^q );

    p=1; q=1;
```

```

printf(" if p=%d , q=%d is p&q=%d, p|q=%d, p^q=%d\n ",p,q,p&q,p|q,p^q);

p=1; q=0;

printf(" if p=%d , q=%d is p&q=%d, p|q=%d, p^q=%d\n ",p,q,p&q,p|q,p^q);

getch( );

}

```

### **ខាងក្រោមជាលទ្ធផលកម្មវិធីខាងលើ**

if p=0 , q = 0 is p & q = 0 , p | q = 0 , p^q = 0

if p=0 , q = 1 is p & q = 0 , p | q = 1 , p^q = 1

if p=1 , q = 1 is p & q = 1 , p | q = 1 , p^q = 0

if p=1 , q = 0 is p & q = 0 , p | q = 1 , p^q = 1

ប្រសិន A = 60 and B = 13 នៅក្នុងទំរង់ binary , យើងនឹងទទួលបានលទ្ធផលដូចខាងក្រោម

A = 0011 1100

B = 0000 1101

-----

A&B = 0000 1100

A|B = 0011 1101

A^B = 0011 0001

~A = 1100 0011

តារាងខាងក្រោមរាយបញ្ជីប្រតិបត្តិសញ្ញា bitwise ដែលប្រើក្នុងភាសា C. សន្មតថាអថេរ 'A' ទទួលបានតម្លៃ 60 ហើយអថេរ 'B' ទទួលបានតម្លៃ 13 បន្ទាប់មក –

| Operator | Description | Example |
|----------|-------------|---------|
|----------|-------------|---------|

|    |                                                                                                                           |                               |
|----|---------------------------------------------------------------------------------------------------------------------------|-------------------------------|
| &  | Binary AND Operator copies a bit to the result if it exists in both operands.                                             | (A & B) = 12, i.e., 0000 1100 |
|    | Binary OR Operator copies a bit if it exists in either operand.                                                           | (A   B) = 61, i.e., 0011 1101 |
| ^  | Binary XOR Operator copies the bit if it is set in one operand but not both.                                              | (A ^ B) = 49, i.e., 0011 0001 |
| ~  | Binary One's Complement Operator is unary and has the effect of 'flipping' bits.                                          | (~A) = ~(60), i.e., -0111101  |
| << | Binary Left Shift Operator. The left operands value is moved left by the number of bits specified by the right operand.   | A << 2 = 240 i.e., 1111 0000  |
| >> | Binary Right Shift Operator. The left operands value is moved right by the number of bits specified by the right operand. | A >> 2 = 15 i.e., 0000 1111   |

## 5. Assignment Operators

តារាងខាងក្រោមបង្ហាញពីការប្រើប្រាស់សញ្ញាស្នើក្នុងការធ្វើប្រមាណវិធីនិងទិន្នន័យ

| Operator | ការពិពណ៌នា                                                                                                            | Example                                       |
|----------|-----------------------------------------------------------------------------------------------------------------------|-----------------------------------------------|
| =        | Simple assignment operator. Assigns values from right side operands to left side operand                              | C = A + B will assign the value of A + B to C |
| +=       | Add AND assignment operator. It adds the right operand to the left operand and assign the result to the left operand. | C += A is equivalent to C = C + A             |



|     |                                                                                                                                     |                                           |
|-----|-------------------------------------------------------------------------------------------------------------------------------------|-------------------------------------------|
| -=  | Subtract AND assignment operator. It subtracts the right operand from the left operand and assigns the result to the left operand.  | $C -= A$ is equivalent to $C = C - A$     |
| *=  | Multiply AND assignment operator. It multiplies the right operand with the left operand and assigns the result to the left operand. | $C *= A$ is equivalent to $C = C * A$     |
| /=  | Divide AND assignment operator. It divides the left operand with the right operand and assigns the result to the left operand.      | $C /= A$ is equivalent to $C = C / A$     |
| %=  | Modulus AND assignment operator. It takes modulus using two operands and assigns the result to the left operand.                    | $C \% = A$ is equivalent to $C = C \% A$  |
| <<= | Left shift AND assignment operator.                                                                                                 | $C <<= 2$ is same as $C = C << 2$         |
| >>= | Right shift AND assignment operator.                                                                                                | $C >>= 2$ is same as $C = C >> 2$         |
| &=  | Bitwise AND assignment operator.                                                                                                    | $C \&= 2$ is same as $C = C \& 2$         |
| ^=  | Bitwise exclusive OR and assignment operator.                                                                                       | $C \wedge= 2$ is same as $C = C \wedge 2$ |
| =   | Bitwise inclusive OR and assignment operator.                                                                                       | $C  = 2$ is same as $C = C   2$           |

## មេរៀនទី៤      ការប្រើប្រាស់មធ្យមបញ្ចូលនិងបញ្ចេញទិន្នន័យ

### I.    ការប្រើបញ្ចូល Printf

printf ជាបញ្ញាប្រើសំរាប់បង្ហាញទិន្នន័យ។ ហើយមានទំរង់ក្នុងការប្រើដូចខាងក្រោម

Syntax:

```
printf( "Format_List",Arg1,Arg2,... );
```

- Format\_List: ជាទិន្នន័យដែលត្រូវបង្ហាញ
- Arg1,Arg2: ជាអ្នកទុកតម្លៃដែលត្រូវចូលរួមក្នុងការបង្ហាញ

Ex1:

```
printf( "Hello! RUPP" );
```

Ex:2:

```
int a = 5, b=10;
```

```
printf( "value a= %d value b= %d" , a, b );
```

### ខាងក្រោមជាឧទាហរណ៍បង្ហាញពីការប្រើទី១

```
#include<stdio.h>
```

```
#include<conio.h>
```

```
char person_name[20]="Dara kok";
```

```
char sex = 'M';
```

```
int age = 18;
```

```
float salary=13.5;
```

```
void main( ){
```

```
clrscr( );
```

```
printf( "The Staff have Name: %s\n", person_name );
```

```
printf( "The Staff have Sex: %c\n", sex );
```

```
printf( "The Staff have Age: %d\n", age );
```

```

printf( "The Staff have Salary: %d$\n", salary );

getch( );

}

```

### **ខាងក្រោមជាលទ្ធផលកម្មវិធីខាងលើ**

The Staff have Name: Dara kok

The Staff have Sex: M

The Staff have Age: 18

The Staff have Salary: 13.5\$

### **ខាងក្រោមជាឧទាហរណ៍បង្ហាញពីការប្រើទី២**

```

#include<stdio.h>

#include<conio.h>

char person_name[20]="Dara kok";

char sex = 'M';

int age = 18;

float salary=13.5;

void main( ){

    clrscr( );

    printf( "%20s%10s%10s%10s\n", "Person_Name","Sex","Age","Salary");

    printf( "%20s%10c%10d%10.2f\n",person_name,sex,age,salary );

    getch( );

}

```

### **ខាងក្រោមជាលទ្ធផលកម្មវិធីខាងលើ**

| Person_Name | Sex | age | salary |
|-------------|-----|-----|--------|
| Dara kok    | M   | 18  | 13.5   |

## II. ការប្រើបញ្ជា scanf

scanf ជាបញ្ជាប្រើសំរាប់បញ្ចូលទិន្នន័យតាមរយៈ Keyboard ។ ហើយមានទំរង់ក្នុងការប្រើដូចខាងក្រោម

Syntax:

```
scanf( "Format_List",&Arg1,... );
```

- Format List: ជាការយល់ដឹងនៃតម្លៃដែលយើងត្រូវបញ្ចូលទិន្នន័យ
- & : ជានិមិត្តសញ្ញាអាសយដ្ឋាន
- Arg1: ជាអ្នកទុកតំលៃមួយដែលចាំទទួលតម្លៃពី Keyboard ដែលយើងនឹងបញ្ចូលទៅ

Ex1:

```
int a;

scanf( "%d",&a );
```

Ex2:

```
int a;

float t;

scanf( "%d%f",&a,&t );
```

## ខាងក្រោមជាឧទាហរណ៍បង្ហាញពីការប្រើទី១

```
#include<stdio.h>

#include<conio.h>

char person_name[20];

char sex;

int age;

float salary;

void main( ){
```

```

clrscr( );

printf( "Enter Staff Name: ~");

scanf( "%S", &person_name );

printf( "Enter Staff Sex: ~");

scanf( "%C",&sex );

printf( "Enter Staff Age: ~");

scanf( "%d", &age );

printf( "Enter Staff Salary: $ ~");

scanf( "%f", &salary );

printf( "The Result Input Above is: \n");

printf( "The Staff have Name: %s\n", person_name );

printf( "The Staff have Sex: %c\n", sex );

printf( "The Staff have Age: %d\n", age );

printf( "The Staff have Salary: %.2f$\n", salary );

getch( );

}

```

### **ខាងក្រោមជាន់ឧទាហរណ៍កម្មវិធីខាងលើ**

Enter Staff Name: Bopha\_kok

Enter Staff Sex: F

Enter Staff Age: 18

Enter Staff Salary: \$ 14.0

The Staff have Name: Bopha\_kok

The Staff have Sex: F

The Staff have Age: 18

The Staff have Salary: 14.00\$

## **ខាងក្រោមជាឧទាហរណ៍បង្ហាញពីការប្រើទី២**

```
#include<stdio.h>
```

```
#include<conio.h>
```

```
char person_name[20];
```

```
char sex;
```

```
int age;
```

```
float salary;
```

```
void main( ){
```

```
    clrscr( );
```

```
    printf( "Enter Staff Name: " );
```

```
    scanf( "%s", &person_name );
```

```
    printf( "Enter Staff Sex: " );
```

```
    scanf( "%c",&sex );
```

```
    printf( "Enter Staff Age: " );
```

```
    scanf( "%d", &age );
```

```
    printf( "Enter Staff Salary: $ " );
```

```
    scanf( "%f", &salary );
```

```
    printf( "The Result Input Above is: \n" );
```

```
    printf( "%20s%10s%10s%10s\n", "Person_Name", "Sex", "Age", "Salary" );
```

```

printf( "%20s%10c%10d%10.2f$\n",person_name,sex,age,salary );

getch( );

}

```

### ខាងក្រោមជាលទ្ធផលកម្មវិធីខាងលើ

Enter Staff Name: Bopha\_kok

Enter Staff Sex: F

Enter Staff Age: 18

Enter Staff Salary: \$ 14.0

| Person_Name | Sex | age | salary  |
|-------------|-----|-----|---------|
| Bopha_kok   | F   | 18  | 14.00\$ |

### III. ការប្រើបញ្ជា Textcolor

textcolor ជាឃ្លាបញ្ជាប្រើសំរាប់កំណត់ពណ៌អក្សរក្នុងការបង្ហាញ ។ ហើយមានទំរង់ក្នុងការប្រើដូចខាងក្រោម  
Syntax:

```
textcolor( Color );
```

- Color: ជាតម្លៃនៃពណ៌ដែលអាចប្រើបាននៅចន្លោះពី 0 ទៅ 15

Ex:

```
textcolor( 14 ); // 14 ជាតម្លៃនៃពណ៌លឿង (YELLOW)
```

```
cprintf( " Hello! Rupp" );
```

### ខាងក្រោមជាតារាងតម្លៃនៃពណ៌

| COLORS ( text mode ) |       |           |           |
|----------------------|-------|-----------|-----------|
| Constant             | Value | Back-grnd | Fore-grnd |
| BLACK                | 0     | Yes       | Yes       |
| BLUE                 | 1     | Yes       | Yes       |
| GREEN                | 2     | Yes       | Yes       |
| CYAN                 | 3     | Yes       | Yes       |

|              |     |     |     |
|--------------|-----|-----|-----|
| RED          | 4   | Yes | Yes |
| MAGENTA      | 5   | Yes | Yes |
| BROWN        | 6   | Yes | Yes |
| LIGHTGRAY    | 7   | Yes | Yes |
| DARKGRAY     | 8   | No  | Yes |
| LIGHTBLUE    | 9   | No  | Yes |
| LIGHTGREEN   | 10  | No  | Yes |
| LIGHTCYAN    | 11  | No  | Yes |
| LIGHTRED     | 12  | No  | Yes |
| LIGHTMAGENTA | 13  | No  | Yes |
| YELLOW       | 14  | No  | Yes |
| WHITE        | 15  | No  | Yes |
| BLINK        | 128 | No  | No  |

### ខាងក្រោមជាលំហាត់អនុវត្តក្នុងការប្រើ

```
#include<stdio.h>
```

```
#include<conio.h>
```

```
char pro_name[20];
```

```
int qty;
```

```
float price;
```

```
void main( ){
```

```
    clrscr( );
```

```
    textcolor( RED );
```

```
    printf( "Enter Product Name: " );
```

```
    scanf( "%s", &pro_name );
```

```
    textcolor( GREEN );
```



```

        printf( "Enter Product Quantity: ");

        scanf( "%d",&qty );

        textcolor( BLUE );

        printf( "Enter Product Price : ");

        scanf( "%f", &price );

        textcolor( YELLOW );

        printf( "The Result Input Above is: \n" );

        printf( "%20s%10s%10s%10s\n", "Pro_Name","Qty","Price","Amount" );

        printf( "%20s%10c%10.2f%10.2f$\n",pro_name,qty,price,qty * price );

        getch( );

    }

```

### ខាងក្រោមជារូបភាពកម្មវិធីដែលបានរង

Enter Product Name: cocacola

Enter Product Quantity: 5

Enter Product Price: 0.8

| Pro_Name | Qty | Price | Amount  |
|----------|-----|-------|---------|
| cocacola | 5   | 0.80  | 14.00\$ |

#### IV. ការប្រើប្រាស់ TextBackground

textbackground ជាបញ្ញត្តិប្រើសំរាប់កំណត់ពណ៌ផ្ទៃខាងក្រោយអក្សរក្នុងការបង្ហាញ ។ ហើយមានទំរង់ក្នុងការប្រើដូចខាងក្រោម

Syntax:

```
textbackground( Color );
```

- Color: ជាតម្លៃនៃពណ៌ដែលអាចប្រើបាននៅចន្លោះពី 0 ទៅ 15

Ex:

```
textbackground( 1 ); // 1 ជាតម្លៃនៃពណ៌លឿង ( BLUE )
```

```
cprintf( " Hello! Rupp" );
```

### **ខាងក្រោមជាលំហាត់អនុវត្តក្នុងការប្រើ**

```
#include<stdio.h>
```

```
#include<conio.h>
```

```
char pro_name[20];
```

```
int qty;
```

```
float price;
```

```
void main( ){
```

```
    textbackground( BLUE );
```

```
    clrscr( );
```

```
    textbackground( RED );
```

```
    textcolor( CRAY );
```

```
    cprintf( "Enter Product Name: " );
```

```
    scanf( "%s", &pro_name );
```

```
    textcolor( GREEN );
```

```
    cprintf( "Enter Product Quantity: " );
```

```
    scanf( "%d",&qty );
```

```
    textcolor( WHITE );
```

```
    cprintf( "Enter Product Price : " );
```

```
    scanf( "%f", &price );
```

```

        textcolor( YELLOW );

        printf( "The Result Input Above is: \n" );

        printf( "%20s%10s%10s%10s\n", "Pro_Name", "Qty", "Price", "Amount" );

        printf( "%20s%10c%10.2f%10.2f$\n", pro_name, qty, price, qty * price );

        getch( );

    }

```

### ខាងក្រោមជាលទ្ធផលកម្មវិធីខាងលើ

Enter Product Name: cocacola

Enter Product Quantity: 5

Enter Product Price: 0.8

| Pro_Name | Qty | Price | Amount  |
|----------|-----|-------|---------|
| cocacola | 5   | 0.80  | 14.00\$ |

### V. ការប្រើបញ្ជី gotoxy

gotoxy ជាបញ្ជីប្រើសំរាប់កំណត់ទីតាំងក្នុងការបង្ហាញ ។ ហើយមានទំរង់ក្នុងការប្រើដូចខាងក្រោម

Syntax:

```
gotoxy( X, Y );
```

- X: ជាតម្លៃបន្ទាត់ដេកកំណត់ក្នុងការបង្ហាញមានតម្លៃចន្លោះពី 1 ទៅ 80
- Y: ជាតម្លៃបន្ទាត់ឈរកំណត់ក្នុងការបង្ហាញមានតម្លៃចន្លោះពី 1 ទៅ 25

Ex:

```
gotoxy( 1,13 );
```

```
printf( "Welcome to Gotoxy" );
```

### ខាងក្រោមជាលំហាត់អនុវត្តក្នុងការប្រើ

```
#include<stdio.h>
```

```
#include<conio.h>
```

```
char person_name[20];

char sex;

int age;

float salary;

void main( ){

    clrscr( );

    gotoxy( 1,10 );

    printf( "Enter Staff Name: ~ ");

    gotoxy( 30,10 );

    printf( "Enter Staff Sex: ~ ");

    gotoxy( 45,10 );

    printf( "Enter Staff Age: ~ ");

    gotoxy( 60,10 );

    printf( "Enter Staff Salary: $ ~ ");

    gotoxy( 1,11 );

    scanf( "%s", &person_name );

    gotoxy( 30,11 );

    scanf( "%c",&sex);

    gotoxy( 45,11 );

    scanf( "%d", &age );

    gotoxy( 60,11 );

    scanf( "%f", &salary );
```

```

        getch( );
    }

```

### ខាងក្រោមជាលទ្ធផលកម្មវិធីខាងលើ

|                   |                  |                  |                        |
|-------------------|------------------|------------------|------------------------|
| Enter Staff Name: | Enter Staff Sex: | Enter Staff Age: | Enter Staff Salary: \$ |
| Bopha_kok         | F                | 18               | 14.0                   |

### VI. ការប្រើបញ្ជា Delay

delay ជាបញ្ជាប្រើសំរាប់ពន្យារពេលអនុវត្តរបស់ CPU ក្នុងការអនុវត្តពាក្យបញ្ជា ។ ហើយមានទំរង់ក្នុងការប្រើដូចខាងក្រោម

Syntax:

```
delay( Number );
```

- Number: ជាតម្លៃសំរាប់កំនត់ការពន្យារពេលដែលរៀបចំនិងម៉ោង 1000 ប្រហែលនឹង 1second

Ex:      delay( 1000 );

```
printf( "Hello! Delay" );
```

```
delay( 2000 );
```

```
printf( "Bye Bye" );
```

### ខាងក្រោមជាលំហាត់អនុវត្តក្នុងការប្រើ

```
#include<stdio.h>
```

```
#include<conio.h>
```

```
Void main( ){
```

```
    clrscr( );
```

```
    delay( 1000 );
```

```
    gotoxy( 37,1 );
```

```
    printf( "Start 1" );
```

```
    delay( 1000 );
```

```

gotoxy( 74,13);

printf( "Start 2" );

delay( 1000);

gotoxy( 37,25 );

printf( "Start 3" );

delay( 1000);

gotoxy( 1,13);

printf( "Start 4" );

delay( 1000);

gotoxy( 37,13);

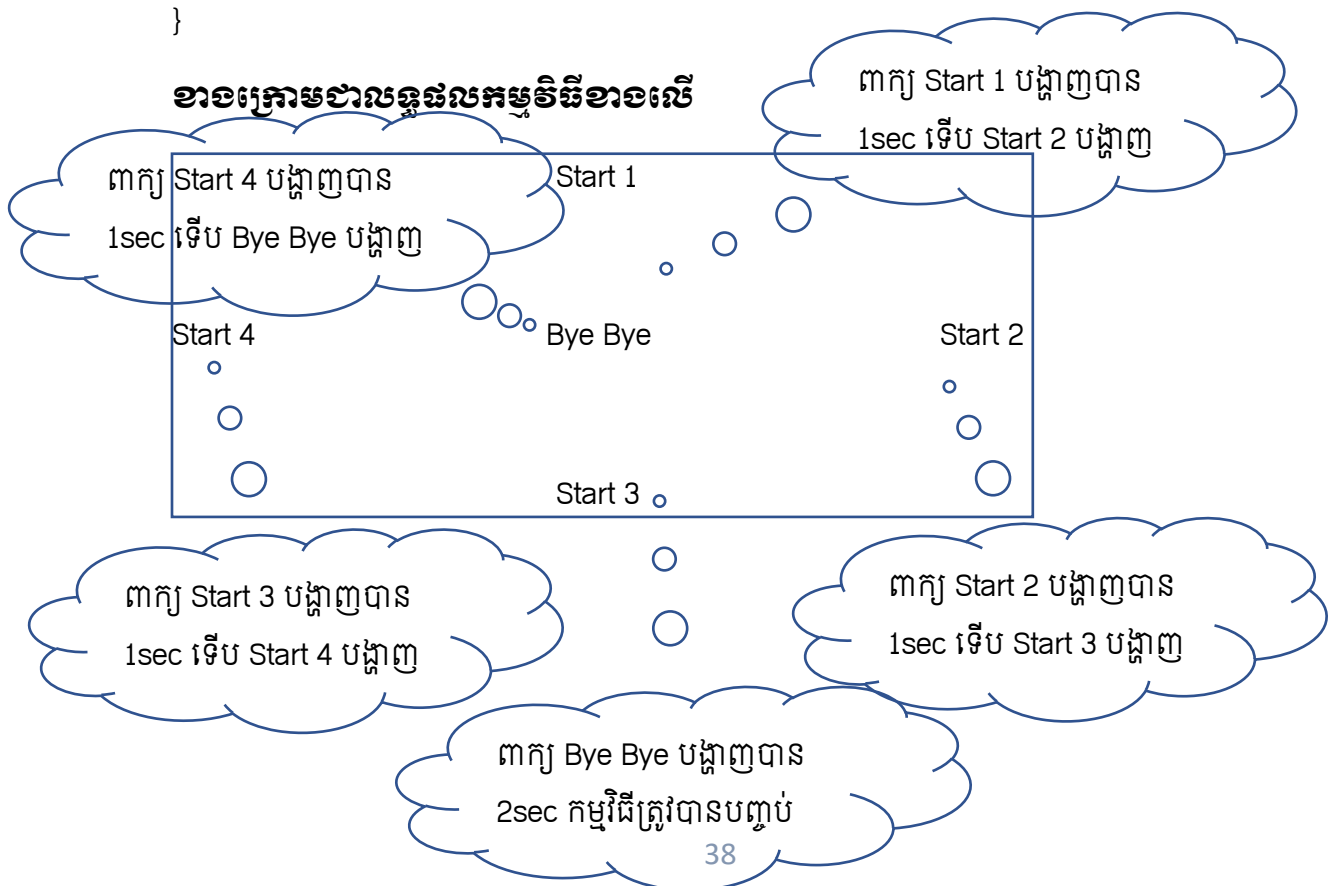
printf( "Bye Bye" );

delay( 2000);

}

```

### ខាងក្រោមជារូបភាពអង្គការនីមួយៗ



## VII. ការប្រើប្រាស់ Clreol

clreol ជាបញ្ញាប្រើសំរាប់លប់ទិន្នន័យក្រោយ Cursor មួយបន្ទាត់ ។ ហើយមានទំរង់ក្នុងការប្រើដូចខាងក្រោម

Syntax:

```
clreol( );
```

Ex:

```
gotoxy( 1,1);
```

```
clreol( );
```

### ខាងក្រោមជាលំហាត់អនុវត្តក្នុងការប្រើ

```
#include<stdio.h>
```

```
#include<conio.h>
```

```
Void main( ){
```

```
clrscr( );
```

```
gotoxy( 37,1);
```

```
printf( "Start 1" );
```

```
gotoxy( 74,13);
```

```
printf( "Start 2" );
```

```
gotoxy( 37,25);
```

```
printf( "Start 3" );
```

```
gotoxy( 1,13);
```

```
printf( "Start 4" );
```

```
gotoxy( 37,13);
```

```
printf( "Bye Bye" );
```

```
delay( 1000 );
```

```
gotoxy( 37,1);
```

```

    clreol( );

    delay( 1000);

    gotoxy( 74,13);

    clreol( );

    delay( 1000);

    gotoxy( 37,25 );

    clreol( );

    delay( 1000);

    gotoxy( 1,13);

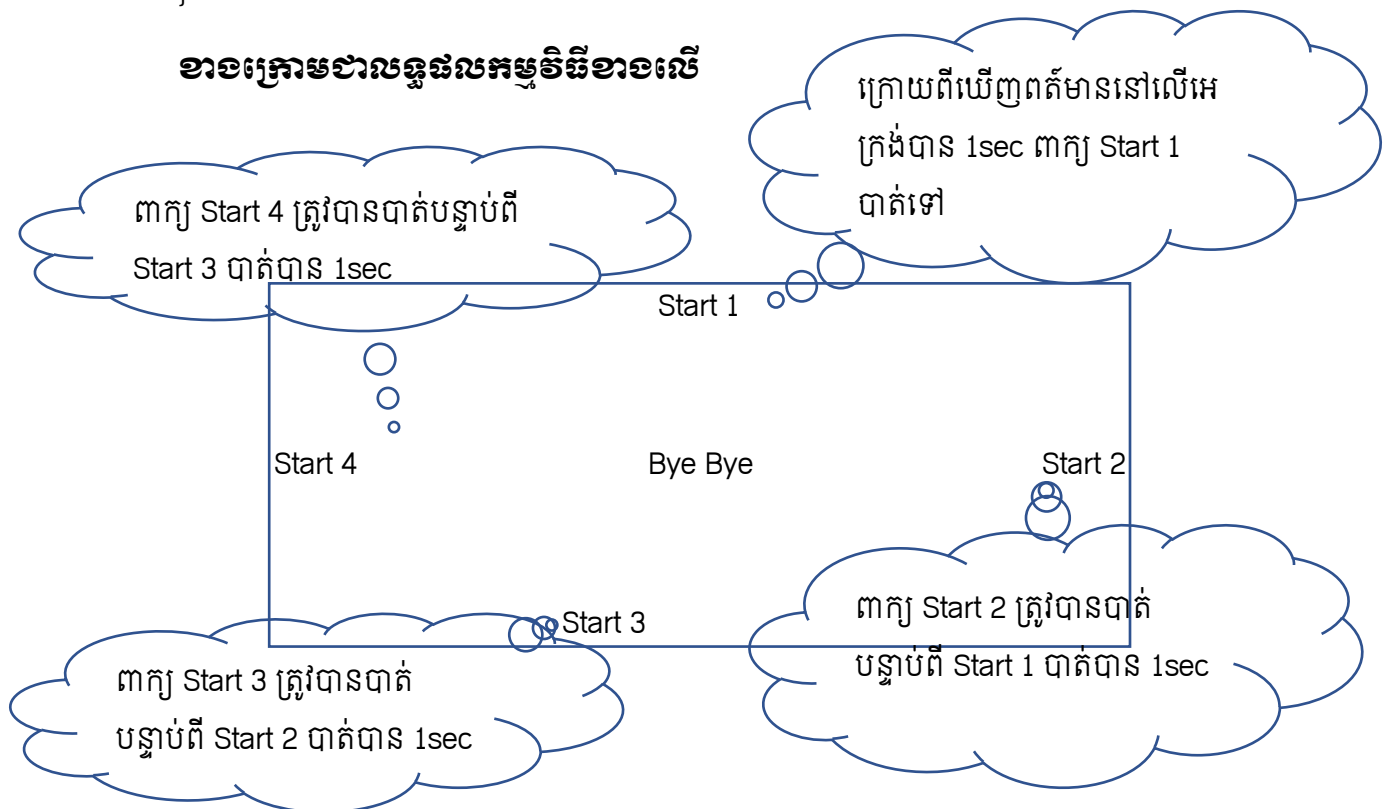
    printf( "%7c", ' ');

    delay( 2000);

}

```

### ខាងក្រោមជាលទ្ធផលកម្មវិធីដែលបានរង





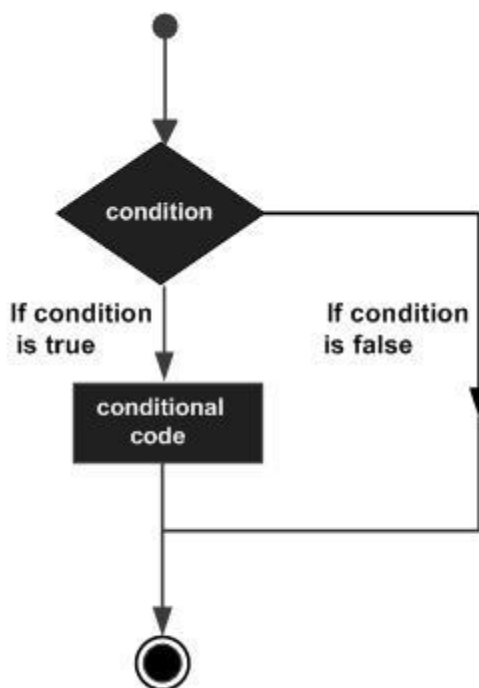
### VIII. លំហាត់អនុវត្ត

ចូរបង្កើតកម្មវិធីអោយអក្សរពាក្យ “Welcome” ផ្លាស់ទីពីទីតាំងខាងធ្វេងទៅស្តាំដែនអាក្រុងបន្ទាត់ទី១ ( $x=74$ ,  $y=1$ ) មកដល់ទីតាំង ( $x=1$ ,  $y=1$ ) នោះពាក្យ “Welcome” ផ្លាស់ទីពីទីតាំងខាងលើចុះក្រោមអាក្រុងបន្ទាត់ទី២៥ ( $x=1$ ,  $y=25$ ) មកដល់ទីតាំងនោះហើយវាផ្លាស់ទីពីស្តាំទៅធ្វេងវិញទៅទីតាំង ( $x=74$ ,  $y=25$ ) មកដល់ទីតាំងនោះហើយវាផ្លាស់ទីពីក្រោមលើងលើវិញទៅទីតាំង ( $x=74$ ,  $y=1$ ) ពេលដល់ទីតាំងនោះហើយកម្មវិធីត្រូវបានបញ្ចប់។

## មេរៀនទី៥ របៀបសរសេរកូដដោយប្រតិបត្តិតាមលក្ខខណ្ឌ

### I. ការប្រើប្រាស់ប្រតិបត្តិតាមលក្ខខណ្ឌ

ការប្រើប្រាស់ប្រតិបត្តិតាមលក្ខខណ្ឌត្រូវអោយយូបញ្ជាទាំងឡាយដែលនៅក្រោមការគ្រប់គ្រងត្រូវបានអនុវត្ត ទាល់តែលក្ខខណ្ឌកំណត់ពិតប្រាកដបើលក្ខខណ្ឌកំណត់មិនពិតយូបញ្ជាទាំងឡាយដែលនៅក្រោមការគ្រប់គ្រងមិនត្រូវ បានអនុវត្តទេ។ ខាងក្រោមជា Flow chat បង្ហាញអំពីការរៀបចំរបស់ការប្រើប្រាស់ប្រតិបត្តិតាមលក្ខខណ្ឌ



### II. ការប្រើប្រាស់ if

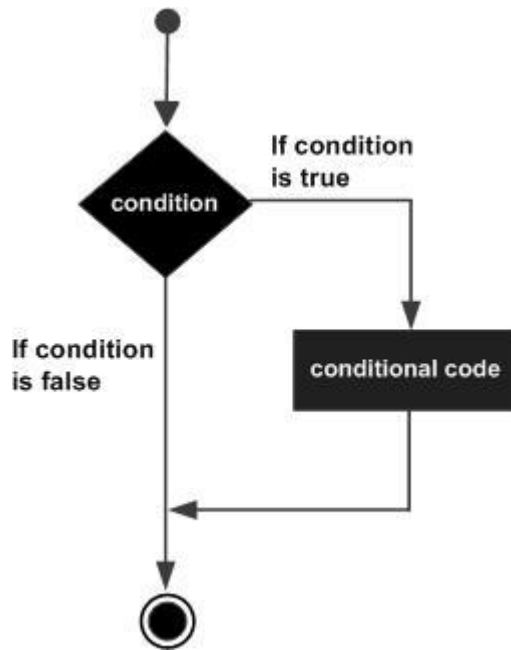
if ជាប្រតិបត្តិនៅពេលដែលមានលក្ខខណ្ឌមួយត្រូវបានគ្រប់គ្រងយូបញ្ជា។ មានទំរង់ទូទៅដូចខាងក្រោម

Syntax:

```

if ( condition )
{
    // statement(s) will execute if the condition is true
}
  
```

**ខាងក្រោមជា Flow chat បញ្ជាក់អំពីការអនុវត្ត**



### ខាងក្រោមជាលំហាត់អនុវត្ត

```

#include <stdio.h>

int main ( ) {
    /* local variable definition */
    int a = 10;
    /* check the boolean condition using if statement */
    if( a < 20 ) {
        /* if condition is true then print the following */
        printf( "a is less than 20\n" );
    }
    printf( "value of a is : %d\n", a );
    return 0;
}
  
```

### ខាងក្រោមជាលទ្ធផលកម្មវិធីខាងលើ

a is less than 20;  
value of a is : 10

### III. ការប្រើប្រាស់ if else

if else ជាឃ្លាប្រតិបត្តិនៅពេលដែលមានលក្ខខណ្ឌមួយត្រូវបានគ្រប់គ្រងឃ្លាបញ្ជាហើយបើលក្ខខណ្ឌមិនពិតមាន ពាក្យបញ្ជាត្រូវបានអនុវត្ត ។ មានទំរង់ទូទៅដូចខាងក្រោម

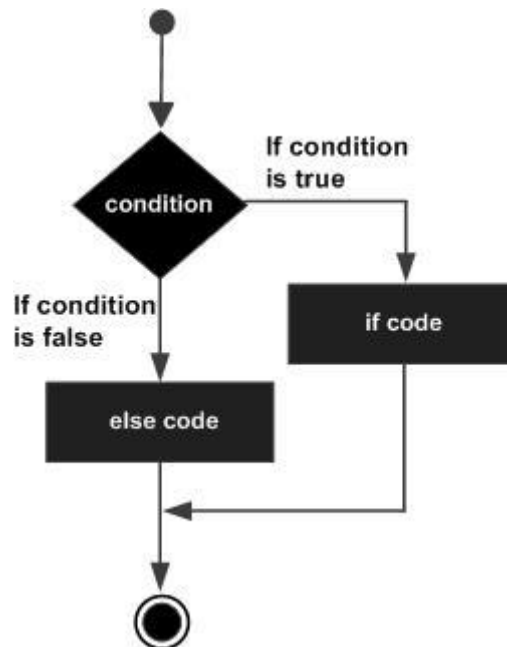
Syntax:

```

if( condition )
{
    /* statement( s ) will execute if the boolean expression is true */
} else {
    /* statement( s ) will execute if the boolean expression is false */
}

```

**ខាងក្រោមជា** Flow chat **បញ្ជាក់អំពីការអនុវត្ត**



**ខាងក្រោមជាលំហាត់អនុវត្ត**

```

#include <stdio.h>

int main ( ) {

    /* local variable definition */

    int a = 100;

```

```

/* check the boolean condition */
if( a < 20 ) {
    /* if condition is true then print the following */
    printf( "a is less than 20\n" );
} else {
    /* if condition is false then print the following */
    printf( "a is not less than 20\n" );
}
printf( "value of a is : %d\n", a );
return 0;
}

```

### **ខាងក្រោមជាលទ្ធផលកម្មវិធីខាងលើ**

a is not less than 20;  
value of a is : 100

Syntax:

```

if( boolean_expression 1 ) {
    /* Executes when the boolean expression 1 is true */
} else if( boolean_expression 2 ) {
    /* Executes when the boolean expression 2 is true */
} else if( boolean_expression 3 ) {
    /* Executes when the boolean expression 3 is true */
} else {
    /* executes when the none of the above condition is true */
}

```

### **ខាងក្រោមជាលំហាត់អនុវត្ត**

```
#include <stdio.h>
```

```
int main ( ) {
```

```
    /* local variable definition */
```

```

int a = 100;

/* check the boolean condition */

if( a == 10 ) {

    /* if condition is true then print the following */

    printf( "Value of a is 10\n" );

} else if( a == 20 ) {

    /* if else if condition is true */

    printf( "Value of a is 20\n" );

} else if( a == 30 )

    {

        /* if else if condition is true */

        printf( "Value of a is 30\n" );

    } else

        {

            /* if none of the conditions is true */

            printf( "None of the values is matching\n" );

        }

printf( "Exact value of a is: %d\n", a );

return 0;

}

```

### **ខាងក្រោមជាន់ផ្តល់លក្ខន្តិកៈនៃខាងលើ**

None of the values is matching

Exact value of a is: 100

#### IV. **ការប្រើប្រាស់** nested if

Nested if ជាឃ្លា if ដែលគេអាចប្រើនៅក្នុងឃ្លា if ឬ if else ។មានទំរង់ទូទៅដូចខាងក្រោម

Syntax:

```
if( boolean_expression 1 )
{
    /* Executes when the boolean expression 1 is true */
    if( boolean_expression 2 )
    {
        /* Executes when the boolean expression 2 is true */
    }
}
```

#### **ខាងក្រោមជាលំហាត់អនុវត្ត**

```
#include <stdio.h>
```

```
int main ( ) {
```

```
    /* local variable definition */
```

```
    int a = 100;
```

```
    int b = 200;
```

```
    /* check the boolean condition */
```

```
    if( a == 100 )
```

```
{
```

```
    /* if condition is true then check the following */
```

```
    if( b == 200 )
```

```
{
```

```
    /* if condition is true then print the following */
```

```
    printf( "Value of a is 100 and b is 200\n" );
```

```
}
```

```

}

printf( "Exact value of a is : %d\n", a );

printf( "Exact value of b is : %d\n", b );

return 0;

}

```

## ខាងក្រោមជាលទ្ធផលកម្មវិធីខាងលើ

Value of a is 100 and b is 200

Exact value of a is : 100

Exact value of b is : 200

## V. ការប្រើប្រាស់ switch

switch ជាប្លង់ប្រតិបត្តិការនៅពេលដែលមានលក្ខខណ្ឌមួយមានជំរើសច្រើនត្រូវបានគ្រប់គ្រងឃ្លាបញ្ជា។ លក្ខខណ្ឌត្រូវតែជាលក្ខខណ្ឌចំៗបើជាតម្លៃលេខត្រូវតែជាលេខគត់បើអក្សរត្រូវតែជាអក្សរមួយហើយត្រូវបានបើដោយពាក្យថា case ។ មានទំរង់ទូទៅដូចខាងក្រោម

Syntax:

```

switch( expression ) {

    case constant-expression :

        statement(s);

        break; /* optional */

    case constant-expression :

        statement(s);

        break; /* optional */

    /* you can have any number of case statements */

    default : /* Optional */

```

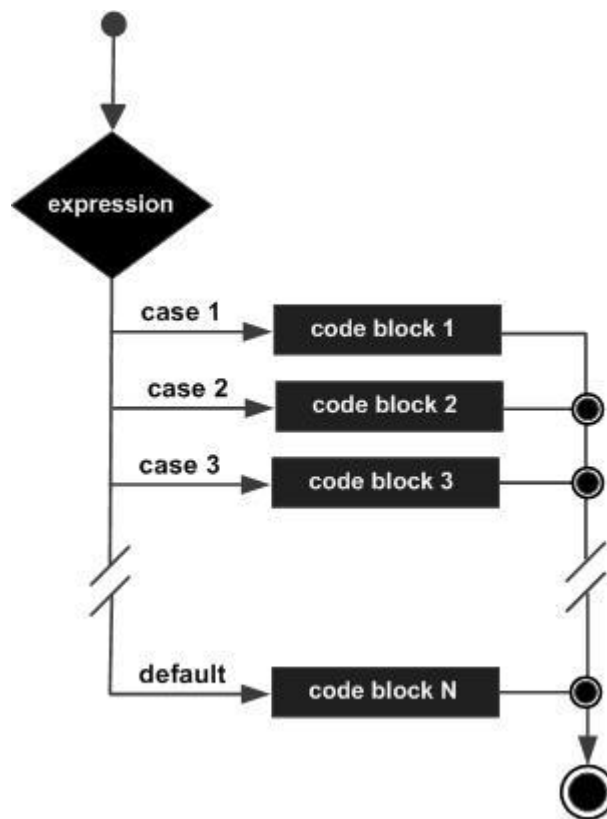


statement(s);

}

- Expression :ជាតម្លៃដែលត្រូវយកមកត្រួតពិនិត្យ
- case :ជាឃ្លាសំរាប់កំណត់តម្លៃដែលត្រូវនិងយកមកត្រួតពិនិត្យ
- constant-expression: តម្លៃត្រូវគ្នានិងតម្លៃយកមកត្រួតពិនិត្យ
- statements : បញ្ជាត្រូវអនុវត្ត
- break; : បញ្ចប់ការងារដែលត្រូវអនុវត្តក្នុងលក្ខខណ្ឌ ( មានក៏បានអត់ក៏បាន )
- default : ជាឃ្លាសំរាប់កំណត់អោយអនុវត្តនៅពេលលក្ខខណ្ឌទាំងអស់មិនពិត ( មានក៏បាន អត់ក៏បាន )

### ខាងក្រោមជា Flow chat បញ្ជាក់អំពីការអនុវត្ត



### ខាងក្រោមជាលំហាត់អនុវត្ត

```
#include <stdio.h>
```

```

int main ( ) {

    /* local variable definition */

    char grade = 'B';

    switch(grade) {

        case 'A' :

            printf( "Excellent!\n" );

            break;

        case 'B' :

        case 'C' :

            printf( "Well done\n" );

            break;

        case 'D' :

            printf( "You passed\n" );

            break;

        case 'F' :

            printf( "Better try again\n" );

            break;

        default :

            printf( "Invalid grade\n" );

    }

    printf( "Your grade is %c\n", grade );

    return 0;

```

}

## ខាងក្រោមជាលទ្ធផលកម្មវិធីខាងលើ

Well done

Your grade is B

### VI. ការប្រើប្រាស់ nested switch

nested switch ជាប្លង់ប្រតិបត្តិក្នុង switch ។ មានទំរង់ទូទៅដូចខាងក្រោម

Syntax:

```
switch( expression ) {  
  
    case constant-expression :  
  
        switch( expression ){  
  
            case constant-expression:  
  
                statement( s );  
  
                break; /* optional */  
  
            ...  
  
            default : /* optional */  
  
                statement( s );  
  
        }  
  
        break; /* optional */  
  
    ...  
  
    default : /* Optional */  
  
        statement( s );  
  
}
```

## ខាងក្រោមជាលំហាត់អនុវត្ត

```
#include <stdio.h>
```

```

int main ( ) {
    /* local variable definition */
    int a = 100;
    int b = 200;
    switch( a ) {
        case 100:
            printf( "This is part of outer switch\n", a );
            switch( b ) {
                case 200:
                    printf( "This is part of inner switch\n", a );
            }
        }
    printf( "Exact value of a is : %d\n", a );
    printf( "Exact value of b is : %d\n", b );
    return 0;
}

```

### **ខាងក្រោមជាលទ្ធផលកម្មវិធីខាងលើ**

This is part of outer switch

This is part of inner switch

Exact value of a is : 100

Exact value of b is : 200

## **VII. ការប្រើសញ្ញា ?**

សញ្ញា ? ជាការកំនត់ប្រតិបត្តិលក្ខខណ្ឌក្នុងឃ្លាបញ្ជា។ វាប្រើមុខងារដូចនឹង if else ដែរហើយមានទំរង់ទូទៅដូចខាងក្រោម

Syntax:

Exp1 ? Exp2 : Exp3;

Ex:

int a = 5, b = 10;

int max = a > b ? a : b;

## ខាងក្រោមជាលំហាត់អនុវត្ត

```
#include<stdio.h>

#include<conio.h>

int a, b, c;

void main( ){

    clrscr( );

    printf( "Enter value a: " );

    scanf( "%d",&a );

    printf( "Enter value b: " );

    scanf( "%d",&b );

    printf( "Enter value c: " );

    scanf( "%d",&c );

    printf( "Maximum value is : %d ", a > b ?

                                                a > c ? a : c :

                                                b > c ? b : c );

    getch( );

}
```

## ខាងក្រោមជាលទ្ធផលកម្មវិធីខាងលើ

Enter value a: 5

Enter value b: 10

Enter value c: 3

Maximum value is : 10

## VIII. ការប្រើប្រាស់ goto

goto ជាប្លង់ប្រើសំរាប់លត់រំលងពាក្យបញ្ជា ។ មានទំរង់ទូទៅដូចខាងក្រោម

Syntax:

```
goto Label_name;  
Statement( s );  
...  
Label_name:
```

Or

```
Label_name:  
Statement( s );  
...  
goto Label_name;
```

## ខាងក្រោមជាលំហាត់អនុវត្ត

```
#include<stdio.h>
```

```
#include<conio.h>
```

```
char pro_name[20];
```

```
int qty;
```

```
float price, amount=0.0;
```

```
void main( ){ int i = 0;
```

```
clrscr( );
```

**again:**

```
gotoxy( 1,1 );
```

```
printf( "Enter Product Name: " );
```

```
clrscr( );
```

```
scanf( "%s", &pro_name );
```

```
printf( "Enter Product Quantity: " );
```

```

    clrerr( );

    scanf( "%d",&qty );

    printf( "Enter Product Price : ");

    clrerr( );

    scanf( "%f", &price );

    printf( "The Result Input Above is: \n" );

    gotoxy( 1,5 );

    printf( "%20s%10s%10s%10s", "Pro_Name","Qty","Price","Amount" );

    gotoxy( 1, 6 + i );

    printf( "%20s%10c%10.2f%10.2f$\n",pro_name,qty,price,qty * price );

    amount += qty * price;

    gotoxy( 1, 7+i ); printf( "press any key to paid or Enter key to buy continue" );

    i++;

    if ( getch( ) == 13 ) goto again;

    gotoxy( 1, 7+i );

    printf( "Amount paid is %.2f$",amount );

    getch( );

}

```

### **ខាងក្រោមជាន់ផ្តល់លក្ខណៈវិធីខាងលើ**

Enter Product Name: cocacola

Enter Product Quantity: 5

Enter Product Price: 0.8

| Pro_Name | Qty | Price | Amount  |
|----------|-----|-------|---------|
| cocacola | 5   | 0.80  | 14.00\$ |

press any key to paid or Enter key to buy continue

Amount paid is : 14.00\$

#### IX. **ការប្រើប្រាស់** exit

exit ជា ឃ្លាប្រើសំរាប់បញ្ចប់កម្មវិធី ។ មានទំរង់ទូទៅដូចខាងក្រោម

Syntax:

```
exit( number );
```

#### **ខាងក្រោមជាលំហាត់អនុវត្ត**

```
#include<stdio.h>
```

```
#include<conio.h>
```

```
int a, b;
```

```
void main( ){
```

```
again:
```

```
clrscr( );
```

```
printf( "Enter value a : " );
```

```
scanf( "%d", &a );
```

```
printf( "Enter value b: " );
```

```
scanf( "%d", &b );
```

```
printf( "Press key to show\n" );
```

```
printf( "1/ Sum a + b\n" );
```

```
printf( "2/ Sub a - b\n" );
```

```
printf( "3/ Mul a * b\n" );
```

```
printf( "4/ Dev a / b\n" );
```

```
printf( "5/ Mol a %b\n" );
```

```
printf( "Press other key exit" );
```

```
switch( getch( ) ){
```

```
case '1' :
```

```
printf( "Sum ( %d + %d ) = %d",a , b , a+ b );
```



```

        break;
    case '2' :
        printf( "Sub ( %d - %d ) = %d",a , b , a- b);
        break;
    case '3' :
        printf( "Mul ( %d * %d ) = %d",a , b , a * b);
        break;
    case '4' :
        printf( "Dev ( %d / %d ) = %.2f",a , b , a/ float(b) );
        break;
    case '5' :
        printf( "Mol ( %d %% %d ) = %d",a , b , a% b);
        break;
    default:      exit( 1 );
}
getch( );
goto again;
}

```

### **ខាងក្រោមជាន់ផ្តល់កម្មវិធីខាងលើ**

Enter value a: 10

Enter value b: 5

Press key to show

1/ Sum a + b

2/ Sub a – b

3/ Mul a \* b

4/ Dev a / b

5/ Mol a % b

Press other to exit // press key 1

Sum ( 10 + 5 ) = 15

X. **ការប្រើប្រាស់** tolower

tolower ជា ឃ្លាប្រើសំរាប់បំប្លែងអក្សរមួយអោយទៅជាអក្សរតូច។ មានទំរង់ទូទៅដូចខាងក្រោម

Syntax:

```
tolower( char );
```

**ខាងក្រោមជាលំហាត់អនុវត្ត**

```
#include<stdio.h>
```

```
#include<conio.h>
```

```
int x=40, y=13;
```

```
char ch;
```

```
void main( ){
```

```
    printf( "Please Press key below to play\n" );
```

```
    printf( "key 'a' play Left\n" );
```

```
    printf( "key 'd' play Right\n" );
```

```
    printf( "key 'w' play Up\n" );
```

```
    printf( "key 's' play Down\n" );
```

```
    printf( "key ESC to exit" );
```

again:

```
    clrscr( );
```

```
    gotoxy( x,y );
```

```
    printf( "A" );
```

```
    ch=getch( );
```

```
    switch( tolower( ch ) ){
```

```
    case 'a' :
```

```
        x = x - 1;
```

```
        if ( x<1 )
```

```
            x = 80;
```

```
        break;
```

```
    case 'd' :
```

```
        x = x + 1;
```

```

        if ( x>80 )

            x = 1;

        break;

case 'w' :

    y = y - 1;

    if ( y<1 )

        y = 25;

    break;

case 's' :

    y = y + 1;

    if ( y>25 )

        y = 1;

    break;

case 27:      exit( 0 );

    }

goto again;

}

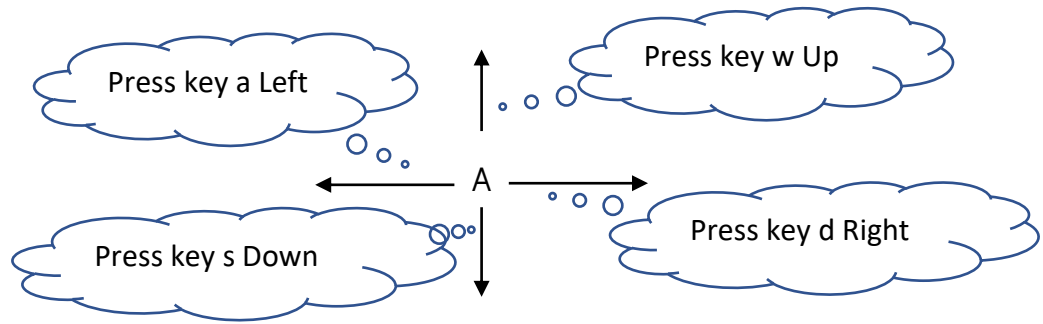
```

## ខាងក្រោមជាលក្ខណៈកម្មវិធីខាងលើ

```

printf( "Please Press key below to play\n" );
printf( "key `a` play Left\n" );
printf( "key `d` play Right\n" );
printf( "key `w` play Up\n" );
printf( "key `s` play Down\n" );
printf( "key ESC to exit" );

```



# XI. ការប្រើប្រាស់ toupper

toupper ជាប្លង់ប្រើសំរាប់បំប្លែងអក្សរមួយអោយទៅជាអក្សរធំ។ មានទំរង់ទូទៅដូចខាងក្រោម

Syntax:

```
toupper( char );
```

## ខាងក្រោមជាលំហាត់អនុវត្ត

```
#include<stdio.h>
```

```
#include<conio.h>
```

```
int x=40, y=13;
```

```
char ch;
```

```
void main( ){
```

```
    printf( "Please Press key below to play\n" );
```

```
    printf( "key 'A' play Left\n" );
```

```
    printf( "key 'D' play Right\n" );
```

```
    printf( "key 'W' play Up\n" );
```

```
    printf( "key 'S' play Down\n" );
```

```
    printf( "key ESC to exit" );
```

again:

```
    clrscr( );
```

```
    gotoxy( x,y );
```

```
    printf( "A" );
```

```
    ch=getch( );
```

```
    switch( toupper( ch ) ){
```

```
        case 'A' :
```

```

        x = x - 1;
        if ( x<1 )

            x = 80;

        break;

    case 'D' :
        x = x + 1;
        if ( x>80 )

            x = 1;

        break;

    case 'W' :
        y = y - 1;
        if ( y<1 )

            y = 25;

        break;

    case 'S' :
        y = y + 1;
        if ( y>25 )

            y = 1;

        break;

    case 27:      exit( 0 );

    }

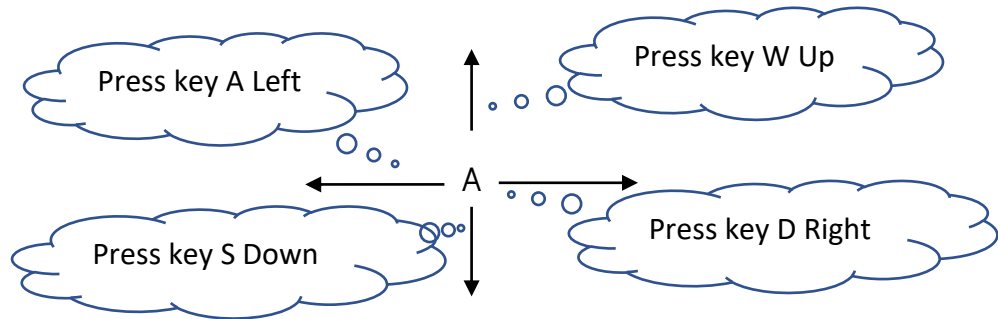
    goto again;

}

```

**ខាងក្រោមជាន់ផ្ទះកម្មវិធីខាងលើ**

```
printf( "Please Press key below to play\n" );
printf( "key `A` play Left\n" );
printf( "key `D` play Right\n" );
printf( "key `W` play Up\n" );
printf( "key `S` play Down\n" );
printf( "key ESC to exit" );
```



## XII. ការប្រើប្រាស់ kbhit

kbhit ជាប្លង់ប្រើសំរាប់ប្រាប់ពីការប៉ះ keyboard បើអ្នកប្រើប៉ះ key ប្រាប់ true(1) បើមិនបានប៉ះ false(0)។  
ខាងក្រោមជាទំរង់ទូទៅ

Syntax:

```
kbhit( );
```

### ខាងក្រោមជាលំហាត់អនុវត្ត

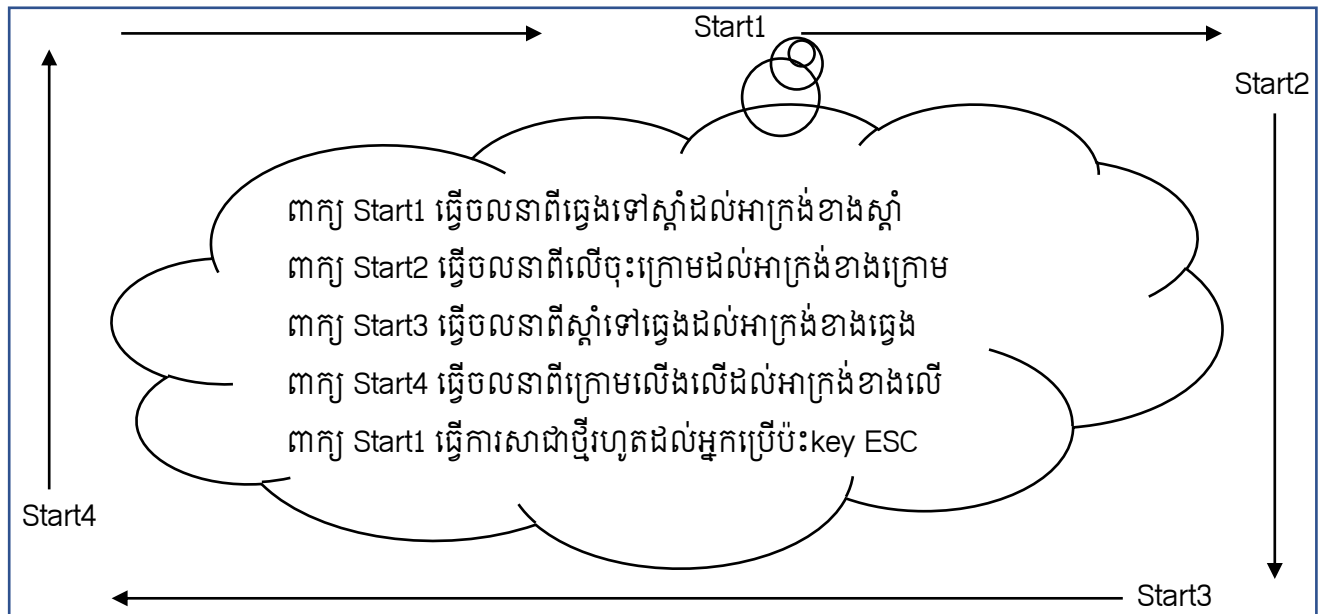
```
#include<stdio.h>
#include<conio.h>
#include<dos.h>
#include<stdlib.h>
int i=1, x=40, y=1;
void main( ){
    _setcursortype(_NOCURS);
again:
    clrscr( );
    gotoxy( x,y);
    printf( "start%d",i);
    delay( 50);
```

```

if( kbhit( ) ) {
    if( getch( )==27) exit( 1 );
}
if( i==1 )
{
    x+=1;
    if( x >80-6 )
        i=2;
}
if( i==2 )
{
    y+=1;
    if( y >=25 )
        i=3;
}
if( i==3 )
{
    x=x-1;
    if( x<=1 )
        i=4;
}
if( i==4 )
{
    y=y-1;
    if( y<=1 )
        i=1;
}
goto again;
}

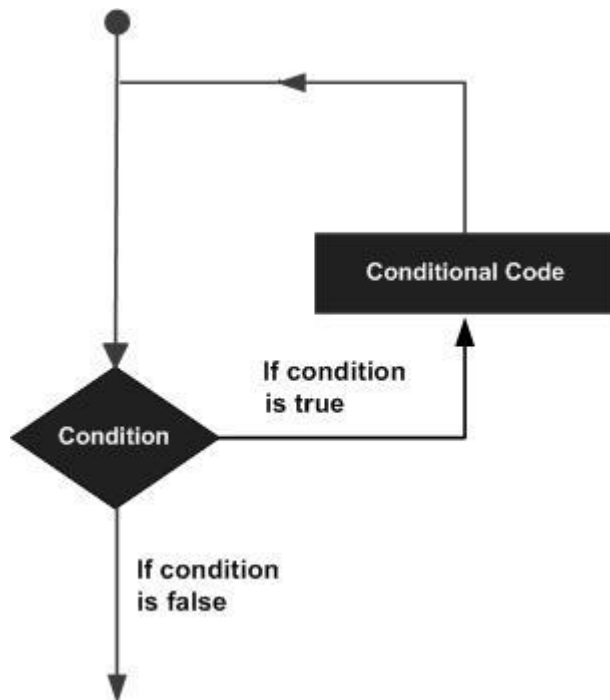
```

**ខាងក្រោមជាលទ្ធផលកម្មវិធីខាងលើ**



### XIII. ការប្រើ control loop

Loop អនុញ្ញាតឱ្យយើងប្រតិបត្តិ statement ឬក្រុមនៃ statements បានច្រើនដង។ ខាងក្រោមគឺជាទម្រង់ទូទៅនៃ Loop statement នៅក្នុងភាសាសរសេរកម្មវិធី





នៅក្នុងភាសា C គេមានរបៀបនៃការប្រើ Loop បីប្រភេទដូចខាងក្រោម

- while
- do while
- for

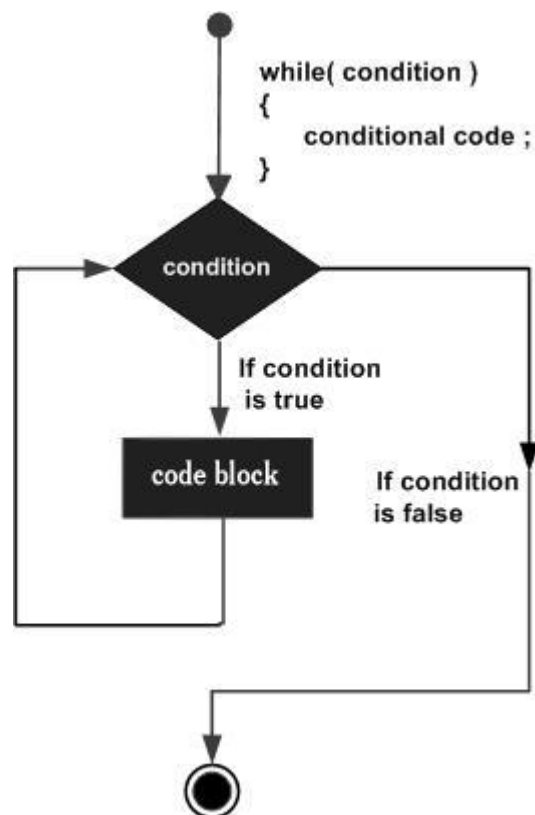
#### XIV. ការប្រើ while

while ជា Loop ដែរអាចអោយ statement អនុវត្តដដែលៗបានច្រើនដងដោយពិនិត្យលក្ខខណ្ឌពិតទើប statement ត្រូវបានអនុវត្ត។ ខាងក្រោមជាទំរង់ទូទៅនៃការប្រើ

Syntax:

```
while( condition )  
{  
    statement( s );  
}
```

**ខាងក្រោមជា** Flow chat **បញ្ជាក់អំពីការអនុវត្ត**



## ខាងក្រោមជាលំហាត់អនុវត្ត

```
#include <stdio.h>

int main ( ) {
    /* local variable definition */
    int a = 10;
    /* while loop execution */
    while( a < 20 )
    {
        printf( "value of a: %d\n", a );
        a++;
    }
    return 0;
}
```

## ខាងក្រោមជាលទ្ធផលកម្មវិធីខាងលើ

value of a: 10

value of a: 11

value of a: 12

value of a: 13

value of a: 14

value of a: 15

value of a: 16

value of a: 17

value of a: 18

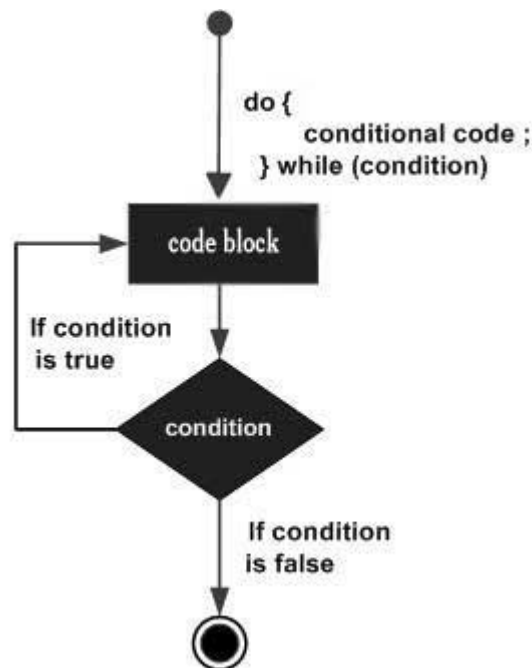
value of a: 19

XV. **ការប្រើ** do while

do while ជា Loop ដែលអាចអោយ statement អនុវត្តដដែលៗបានច្រើនដងដូចទៅនឹង while ដែលតែងតែមានការអនុវត្តខុសគ្នាដោយ statement អនុវត្តរួចទើបពិនិត្យលក្ខខណ្ឌជាក្រោយ។ ខាងក្រោមជាទំរង់ទូទៅនៃការប្រើ Syntax:

```
do
{
    statement(s);
} while( condition );
```

### ខាងក្រោមជា Flow chat បញ្ជាក់អំពីការអនុវត្ត



### ខាងក្រោមជាលំហាត់អនុវត្ត

```
#include <stdio.h>

int main ( ) {
    /* local variable definition */
    int a = 10;
    /* do loop execution */
    do {
```

```

printf( "value of a: %d\n", a );
a = a + 1;
}while( a < 20 );

return 0;
}

```

## ខាងក្រោមជាលទ្ធផលកម្មវិធីខាងលើ

value of a: 10  
 value of a: 11  
 value of a: 12  
 value of a: 13  
 value of a: 14  
 value of a: 15  
 value of a: 16  
 value of a: 17  
 value of a: 18  
 value of a: 19

## XVI. ការប្រើ for

for ជា Loop ដូចទៅ while និង do while ដែរតែគេត្រូវបានប្រើវានៅពេលដែលការងារដែលត្រូវបានអនុវត្តមានចំនួនកំណត់ច្បាស់លាស់ ។ខាងក្រោមជាទំរង់ទូទៅ

Syntax:

```

for ( init; condition; increment )
{
    statement(s);
}

```

Ex:

```

for(int i=0; i<5; i++){

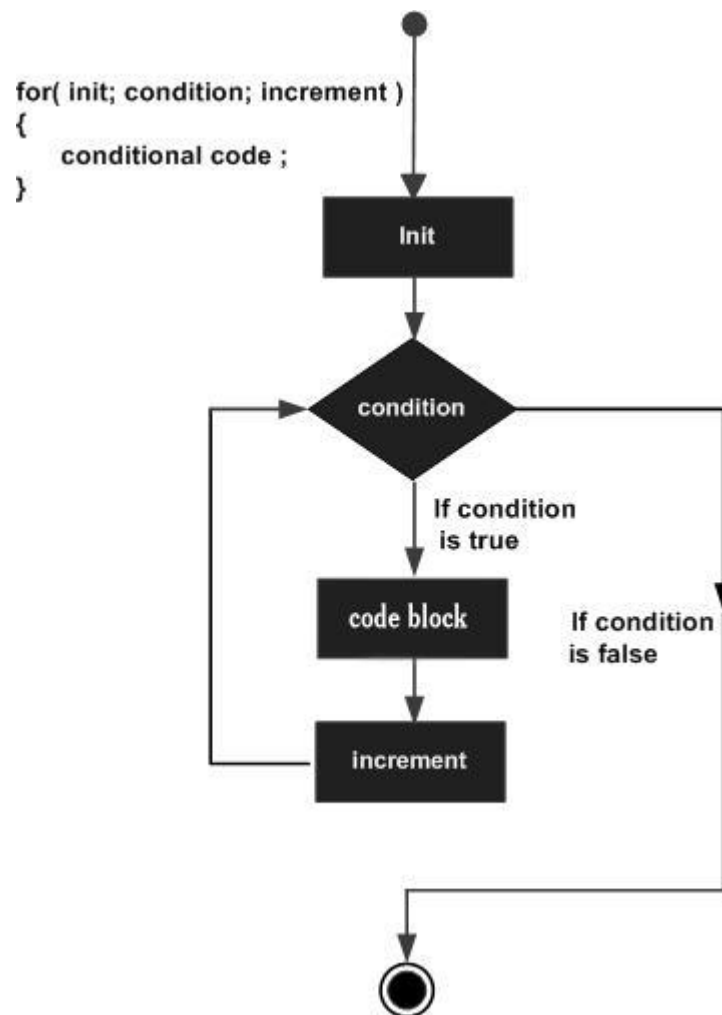
```

```
printf( "%d\n",i);
```

```
}
```

- Init : ជាការកំណត់តម្លៃចាប់ផ្តើម
- Condition : ជាលក្ខខណ្ឌកំណត់
- Increment : ជាអ្នករាប់តម្លៃ

**ខាងក្រោមជា Flow chat បញ្ជាក់អំពីការអនុវត្ត**



**ខាងក្រោមជាលំហាត់អនុវត្ត**

```
#include <stdio.h>
```

```
int main ( ) {
```

```

int a;

/* for loop execution */

for( a = 10; a < 20; a = a + 1 )

{

    printf("value of a: %d\n", a);

}

return 0;

}

```

### **ខាងក្រោមជាលទ្ធផលកម្មវិធីខាងលើ**

value of a: 10  
 value of a: 11  
 value of a: 12  
 value of a: 13  
 value of a: 14  
 value of a: 15  
 value of a: 16  
 value of a: 17  
 value of a: 18  
 value of a: 19

### **XVII. ការប្រើ nested Loop**

Nested Loop ជាការប្រើ Loop នៅក្នុង Loop ។ មានទំរង់ទូទៅដូចខាងក្រោម

Syntax while:

```

while( condition )
{
    while( condition )
    {

```

```

        statement(s);
    }
    statement(s);
}

```

Syntax do while:

```

do {
    statement(s);

do {
    statement(s);
}while( condition );

```

```

}while( condition );

```

Syntax for:

```

for ( init; condition; increment )
{
    for ( init; condition; increment )
    {
        statement(s);
    }
    statement(s);
}

```

**ខាងក្រោមជាលំហាត់អនុវត្ត**

#include <stdio.h>

```

int main ( ) {
    /* local variable definition */
    int i, j;
    for( i = 2; i<100; i++ ) {
        for( j = 2; j <= ( i/j ); j++ )
            if( !( i%j ) ) break; // if factor found, not prime
        if( j > ( i/j ) ) printf( "%d is prime\n", i );
    }
    return 0;
}

```

### **ខាងក្រោមជាលទ្ធផលកម្មវិធីខាងលើ**

2 is prime  
 3 is prime  
 5 is prime  
 7 is prime  
 11 is prime  
 13 is prime  
 17 is prime  
 19 is prime  
 23 is prime  
 29 is prime  
 31 is prime  
 37 is prime  
 41 is prime  
 43 is prime  
 47 is prime  
 53 is prime  
 59 is prime  
 61 is prime  
 67 is prime  
 71 is prime



73 is prime  
79 is prime  
83 is prime  
89 is prime  
97 is prime

### XVIII. ការប្រើ break

break ជា keyword ប្រើសំរាប់បញ្ចប់ការងាររបស់ Loop ។ ខាងក្រោមជាឧទាហរណ៍នៃការប្រើ

int i=1;

Ex1:

```
#include<stdio.h>
```

```
#include<conio.h>
```

```
void main( ){
```

```
do{
```

```
    textcolor(i);
```

```
    gotoxy( 33,13 );
```

```
    printf( "Welcome to RUPP" );
```

```
    i++;
```

```
    if( i>15 ) i=1;
```

```
    if( kbhit( ) ) break;
```

```
}while( 1 );
```

```
}
```

**ខាងក្រោមជាលទ្ធផលកម្មវិធីខាងលើ**

Welcome to RUPP      // ពាក្យ Welcome to RUPP ត្រូវបានបង្ហាញដោយមានការផ្លាស់ប្តូរពណ៌រហូត  
// ទាល់តែមានការប៉ះ key ណាមួយទើបកម្មវិធីត្រូវបានបញ្ចប់

Ex2:

```
#include<stdio.h>
```

```
#include<conio.h>
```

```
void main( ){
```

```

while( 1 ){
    textcolor(i);
    gotoxy( 33,13 );
    cprintf( "Welcome to RUPP" );
    i++;
    if( i>15 ) i=1;
    if( kbhit( ) ) break;
}
}

```

### **ខាងក្រោមជាលក្ខណៈកម្មវិធីខាងលើ**

Welcome to RUPP      // ពាក្យ Welcome to RUPP ត្រូវបានបង្ហាញដោយមានការផ្លាស់ប្តូរពណ៌រហូត  
 // ទាល់តែមានការប៉ះ key ណាមួយទើបកម្មវិធីត្រូវបានបញ្ចប់

Ex3:

```

#include<stdio.h>
#include<conio.h>
void main( ){
    for( ; ; ){
        textcolor(i);
        gotoxy( 33,13 );
        cprintf( "Welcome to RUPP" );
        i++;
        if( i>15 ) i=1;
        if( kbhit( ) ) break;
    }
}

```

### **ខាងក្រោមជាលក្ខណៈកម្មវិធីខាងលើ**

```
Welcome to RUPP // ពាក្យ Welcome to RUPP ត្រូវបានបង្ហាញដោយមានការផ្លាស់ប្តូរពណ៌រហូត
// ទាល់តែមានការប៉ះ key ណាមួយទើបកម្មវិធីត្រូវបានបញ្ចប់
```

#### XIX. **ការប្រើ** continue

continue ជា keyword ប្រើសំរាប់បន្តត្រួតពិនិត្យលក្ខខណ្ឌ Loop សារជាថ្មី ។ ខាងក្រោមជាឧទាហរណ៍នៃការប្រើ

Ex:

```
#include<stdio.h>

#include<conio.h>

int i;

void main( ){

    clrscr( );

    for( i = 1; i< 20 ; i++ ){

        if( i%2==0 ) continue;

        printf( "Odd value is : %d\n",i);

    }

    getch( );

}
```

#### **ខាងក្រោមជាលទ្ធផលកម្មវិធីខាងលើ**

```
Odd value is : 1
Odd value is : 3
Odd value is : 5
Odd value is : 7
Odd value is : 9
Odd value is : 11
Odd value is : 13
Odd value is : 15
Odd value is : 17
Odd value is : 19
```

## XX. ការប្រើ random

random ជាប្លង់ប្រើសំរាប់ប្រាប់តម្លៃចៃដន្យនៅចន្លោះពី 0 ទៅចំនួនដែលយើងផ្តល់អោយវា។ខាងក្រោមជាទំរង់ទូទៅនៃការប្រើ

Syntax:

```
random( number );
```

Ex:      `int i = random( 5 );    // 0, 1, 2, 3, 4`

## ខាងក្រោមជាលំហាត់អនុវត្ត

```
#include<stdio.h>
```

```
#include<conio.h>
```

```
#include<stdlib.h>
```

```
#include<dos.h>
```

```
int x;
```

```
int y;
```

```
void main( ){
```

```
    clrscr( );
```

```
    do{
```

```
        x= random( 67 )+1; // max x = 80 – 13 character
```

```
        y= random( 25 );    // max y = 25
```

```
        gotoxy( x,y );
```

```
        printf( "welcome to rupp" );
```

```
        delay( 100 );
```

```
        while( !kbhit( ) );
```

```
    }
```

## ខាងក្រោមជាលទ្ធផលកម្មវិធីខាងលើ



## XXI. ការប្រើ randomize

randomize ជាប្លង់ប្រើសំរាប់កំណត់អោយប្លង់ random កុំយកតម្លៃចៃដន្យដដែលៗដំបូងៗ ខាងក្រោមជាទំរង់ ទូទៅនៃការប្រើ

Syntax:

```
randomize( );
```

Ex: randomize( );

```
int i = random(5);
```

## ខាងក្រោមជាលំហាត់អនុវត្ត

```
#include<stdio.h>
```

```
#include<conio.h>
```

```
#include<stdlib.h>
```

```
#include<dos.h>
```

```
int x;
```

```

int y;

void main( ){

    clrscr( );

    randomize( );

    do{

        x= random( 67 )+1; // max x = 80 – 13 character

        y= random( 25 );    // max y = 25

        gotoxy( x,y );

        printf( "welcome to rupp" );

        delay( 100 );

        while( !kbhit( ) );

    }

```

**ខាងក្រោមជាន់ផ្ទាំងកម្មវិធីខាងលើ**

ពាក្យ Welcome to rupp បង្ហាញដោយ  
ចៃដន្យពេលពេញអាក្រុងរហូតទាល់តែ  
អ្នកប្រើប៉ះ Key ណាមួយទើបចប់

Welcome to rupp  
Welcome to rupp

Welcome to rupp

Welcome to rupp

Welcome to rupp

Welcome to rupp

Welcome to rupp

Welcome to rupp

## XXII. លំហាត់បន្ទាញពីការប្ដូរនូវរូបិយប័ណ្ណ

នៅក្នុងលំហាត់ពេលអ្នកប្រើចូលប្រើកម្មវិធីតម្រូវអោយជ្រើសរើសប្រភេទប្រាក់ដែលអ្នកប្រើចង់ប្ដូរចូលរួចហើយ ជ្រើសរើសប្រភេទប្រាក់ដែលត្រូវប្ដូរ

- Riel ទៅ Dollar ឬ Bat
- Dollar ទៅ Riel ឬ Bat
- Bat ទៅ Riel ឬ Dollar

បន្ទាប់មកបញ្ចូលអត្រានៃការប្ដូរនោះអ្នកប្រើនឹងទទួលបានទឹកប្រាក់ដែលនឹងទទួលបាន

**សំគាល់ :** អ្នកប្រើអាចប្ដូរប្រាក់បានច្រើនសារ

```
#include<stdio.h>
#include<conio.h>
#include<stdlib.h>
char ch,st;
float money,rate;
void main( ){
do{
    textbackground( BLUE );
    clrscr( );
    gotoxy( 23,1 );
    textcolor( YELLOW );
    textbackground( RED );
    printf( "Welcome to my program change money" );
    gotoxy( 1,2 );
    textcolor( WHITE );
    textbackground( BLUE );
    printf( "Please select type of money by press key\n\r" );
    printf( "1/Riel\n\r" );
    printf( "2/Dollar\n\r" );
    printf( "3/Bat\n\r" );
again:
    ch=getch( );
    if( ch!='1' && ch!='2' && ch!='3' )
```

```

{
    goto again;
}
printf( "Enter your money %s : ",ch=='1'?"Riel":ch=='2'?"Dollar":"Bat" );
scanf( "%f",&money );
again1:
gotoxy( 1,7 ); clrcl( );
printf( "Current money( %.2f )%c\n",money,ch=='1'?'R':ch=='2'?'$':'B' );
printf( "Select money to change by press key\n" );
switch( ch ){
case '1':
    printf( "1/Dollar\n" );
    printf( "2/Bat\n" );
    printf( "3/chosing again!" );
again2:
    st=getch( );
    if( st!='1' && st!='2' && st!='3' )
        goto again2;
    if( st=='3' ) continue;
    break;
case '2':
    printf( "1/Riel\n" );
    printf( "2/Bat\n" );
    printf( "3/chosing again!" );
again3:
    st=getch( );
    if( st!='1' && st!='2' && st!='3' )
        goto again3;
    if( st=='3' ) continue;
    break;
case '3':
    printf( "1/Riel\n" );
    printf( "2/Dollar\n" );

```



```

        printf( "3/chosing again!");
again4:
    st=getch( );
    if( st!='1' && st!='2' && st!='3')
        goto again4;
    if( st=='3' ) continue;
    break;
}

printf( "\n\rEnter Rate change %s to %s : ",
        ch=='1' ? "Riel":ch=='2' ? "Dollar":"Bat"
        ,ch=='1' ?
        st=='1' ? "Dollar":"Bat" :
        ch=='2' ?
        st=='1' ? "Riel":"Bat" :
        st=='1' ? "Riel":"Dollar"
        );
scanf( "%f",&rate);
money = ch=='1' ? money/rate:
        ch=='2' ? money *rate:
        st=='1' ? money *rate:money/rate;
ch = ch=='1' ? st=='1' ? '2': '3' :
    ch=='2' ? st=='1' ? '1': '3' :
        st=='1' ? '1': '2';
printf( "You can get your money %.2f %s\n",money,
        ch=='1' ? "Riel":
        ch=='2' ? "Dollar":"Bat"    );
printf( "Continue to change press key Enter other exit" );
if( getch( )==13)
{
    gotoxy( 1,12);clreol( );
    gotoxy( 1,13);clreol( );
    gotoxy( 1,14);clreol( );
    goto again1;
}

```

```

    }
    else
        exit( 1 );
}while( 1 );
}

```

## **ខាងក្រោមនេះជាលក្ខណៈកម្មវិធីខាងលើ**

Welcome to my program change money

Please select type of money by press key

1/Riel

2/Dollar

3/Bat

Enter your money Riel : 1000000

Current money( 1000000 )R

Select money to change by press key

1/Dollar

2/Bat

3/chosing again!

Enter Rate change Riel to Dollar: 4000

You can get your money 250.00 Dollar

Continue to change press key Enter other exit

## I. ការបង្កើត Function

Function ជាការកាត់បន្ថយនៃការសរសេរ code កាត់បន្ថយភាពស្មុគស្មាញនិងងាយស្រួលក្នុងការកែកំហុស។  
ខាងក្រោមជាទំរង់ទូទៅនៃការប្រើ

Syntax:

```
return_type function_name( parameter list )
{
    // body of the function
    return value; // optional
}
```

- return\_type : ជាប្រភេទទិន្នន័យរបស់ Function អាចជា void ឬ ជាប្រភេទទិន្នន័យណាមួយ
- function\_name : ជាឈ្មោះរបស់ Function
- parameter : ជាតម្លៃចាំបាច់ត្រូវផ្តល់អោយ Function អាចមិនត្រូវការ
- return : ជាតម្លៃរបស់ Function ក្រោយពីធ្វើការងារចប់អាចមិនមានពេល return\_type ជា void

Ex:

```
/* function returning the max between two numbers */

int max( int num1, int num2 ) {

    /* local variable declaration */

    int result;

    if ( num1 > num2 )

        result = num1;

    else

        result = num2;

    return result;

}
```

## II. របៀបប្រកាស Function

Function អាចប្រកាសនៅខាងលើ main( ) ឬ សរសេរខាងក្រោម main( ) ឬ ក៏នៅក្នុង Header File។ ខាងក្រោមជាទំរង់ការប្រើ

Syntax:

```
Return_type    function_name( parameter_list );
```

Ex1:

```
int max ( int num1, int num2 );
```

Ex2:

```
int max ( int , int );
```

## III. របៀបហៅ Function

Function អ្នកបង្កើតអត្ថន័យទៅតាមការងារដែលអ្នកចង់បានសំរាប់ការហៅមកប្រើដូច្នេះអ្នកត្រូវចេះហៅវាមកប្រើខាងក្រោមជាលំហាត់បង្ហាញការហៅ Function មកប្រើ។

```
#include <stdio.h>
```

```
/* function declaration */
```

```
int max( int num1, int num2 );
```

```
int main ( ) {
```

```
    /* local variable definition */
```

```
    int a = 100;
```

```
    int b = 200;
```

```
    int ret;
```

```
    /* calling a function to get max value */
```

```
    ret = max( a, b );
```

```
    printf( "Max value is : %d\n", ret );
```

```
    return 0;
```

```
}
```

```
/* function returning the max between two numbers */
```

```
int max( int num1, int num2 ) {
```

```
    /* local variable declaration */
```

```
    int result;
```

```
    if ( num1 > num2 )
```

```
        result = num1;
```

```

else
    result = num2;
return result;
}

```

#### IV. **ការប្រើ** Parameter Function

Function ត្រូវការ parameter យើងត្រូវប្រកាសអថេរចាំទទួលការផ្តល់តំលៃតាមប្រភេទដែលត្រូវការ។ អថេរនោះ ហើយថាអថេរផ្លូវការដូចជាការប្រកាសអថេរនៅក្នុង Function ដែរតែវាការប្រើអោយទទួលតម្លៃរាល់ពេលហៅ Function មកប្រើដោយមានរបៀបប្រើពីរយ៉ាង

- Past by value : តម្លៃផ្តល់ទៅអោយ Parameter Function មិនចេះផ្តល់មកវិញទេ
- Past by reference : តម្លៃផ្តល់ទៅអោយ Parameter Function ចេះផ្តល់ត្រឡប់មកវិញ

##### 1. Past Parameter by value

ការប្រើតាម past by value ចម្លងតម្លៃពិតទៅក្នុងប៉ារ៉ាម៉ែត្រផ្លូវការនៃអនុគមន៍។ ក្នុងករណីនេះ ការផ្លាស់ប្តូរដែលបានធ្វើឡើងចំពោះប៉ារ៉ាម៉ែត្រនៅខាងក្នុង Function មិនមានឥទ្ធិពលលើអ្នកដែលផ្តល់តម្លៃមកនោះទេ។ ខាងក្រោមជាការបង្ហាញនៃរបៀបប្រើ

```

#include <stdio.h>

/* function declaration */
void swap(int x, int y);

int main ( ) {
    /* local variable definition */
    int a = 100;
    int b = 200;

    printf( "Before swap, value of a : %d\n", a );
    printf( "Before swap, value of b : %d\n", b );

    /* calling a function to swap the values */
    swap(a, b);

    printf( "After swap, value of a : %d\n", a );
    printf( "After swap, value of b : %d\n", b );

    return 0;
}

void swap(int x, int y) {

```

```

int temp;

temp = x; /* save the value of x */
x = y;   /* put y into x */
y = temp; /* put temp into y */

return;

}

```

### **ខាងក្រោមនេះជាលទ្ធផលកម្មវិធីខាងលើ**

Before swap, value of a : 100

Before swap, value of b : 200

After swap, value of a : 100

After swap, value of b : 200

## **2. Past Parameter by reference**

ការប្រើតាម past by reference ចម្លងតម្លៃពិតទៅក្នុងប៉ារ៉ាម៉ែត្រផ្លូវការនៃអនុគមន៍។ ក្នុងករណីនេះ ការផ្លាស់ប្តូរដែលបានធ្វើឡើងចំពោះប៉ារ៉ាម៉ែត្រនៅខាងក្នុង Function វាធ្វើអោយមានឥទ្ធិពលលើអ្នកដៃផ្តល់តម្លៃមកវិញ។ ខាងក្រោមជាការបង្ហាញនៃរបៀបប្រើ

```

#include <stdio.h>

int main ( ) {
    /* local variable definition */

    int a = 100;
    int b = 200;

    printf( "Before swap, value of a : %d\n", a );
    printf( "Before swap, value of b : %d\n", b );

    /* calling a function to swap the values */
    swap( &a, &b );

    printf( "After swap, value of a : %d\n", a );
    printf( "After swap, value of b : %d\n", b );

    return 0;
}

void swap( int *x, int *y ) {

```

```

int temp;
temp = *x; /* save the value of x */
*x = *y; /* put y into x */
*y = temp; /* put temp into y */
return;
}

```

### **ខាងក្រោមនេះជាលទ្ធផលកម្មវិធីប្តូរតំលៃ**

Before swap, value of a : 100

Before swap, value of b : 200

After swap, value of a : 200

After swap, value of b : 100

## **V. ការបង្កើត Recursion Function**

Recursion Function គឺជា Function ដែលមានការហៅខ្លួនឯងប្រើក្នុងខ្លួនឯង ។ ខាងក្រោមជាទំរង់ទូទៅ

Syntax:

```

void recursion( ) {
    recursion( ); /* function calls itself */
}
int main( ) {
    recursion( );
}

```

### **ខាងក្រោមជាការបង្ហាញនៃការបង្កើត recursive function factorial**

```
#include <stdio.h>
```

```
unsigned long long int factorial(unsigned int i) {
```

```
    if(i <= 1)
```

```
{
```

```
    return 1;
```

```

}

return i * factorial(i - 1);

}

int main( ) {

    int i = 12;

    printf("Factorial of %d is %d\n", i, factorial(i) );

    return 0;

}

```

**ខាងក្រោមជាលទ្ធផលកម្មវិធីខាងលើ**

Factorial of 12 is 479001600

**ខាងក្រោមជាការបង្ហាញនៃការបង្កើត recursive function**

```
include <stdio.h>
```

```
int fibonacci( int i )
```

```

{

    if( i == 0 )

    {

        return 0;

    }

    if( i == 1 )

    {

        return 1;

    }

```



```

return fibonacci(i-1) + fibonacci(i-2);

}

int main( )

{

    int i;

    for ( i = 0; i < 10; i++)

    {

        printf( "%d\\n", fibonacci(i) );

    }

    return 0;

}

```

### ខាងក្រោមជាលទ្ធផលកម្មវិធីខាងលើ

```

0
1
1
2
3
5
8
13
21
34

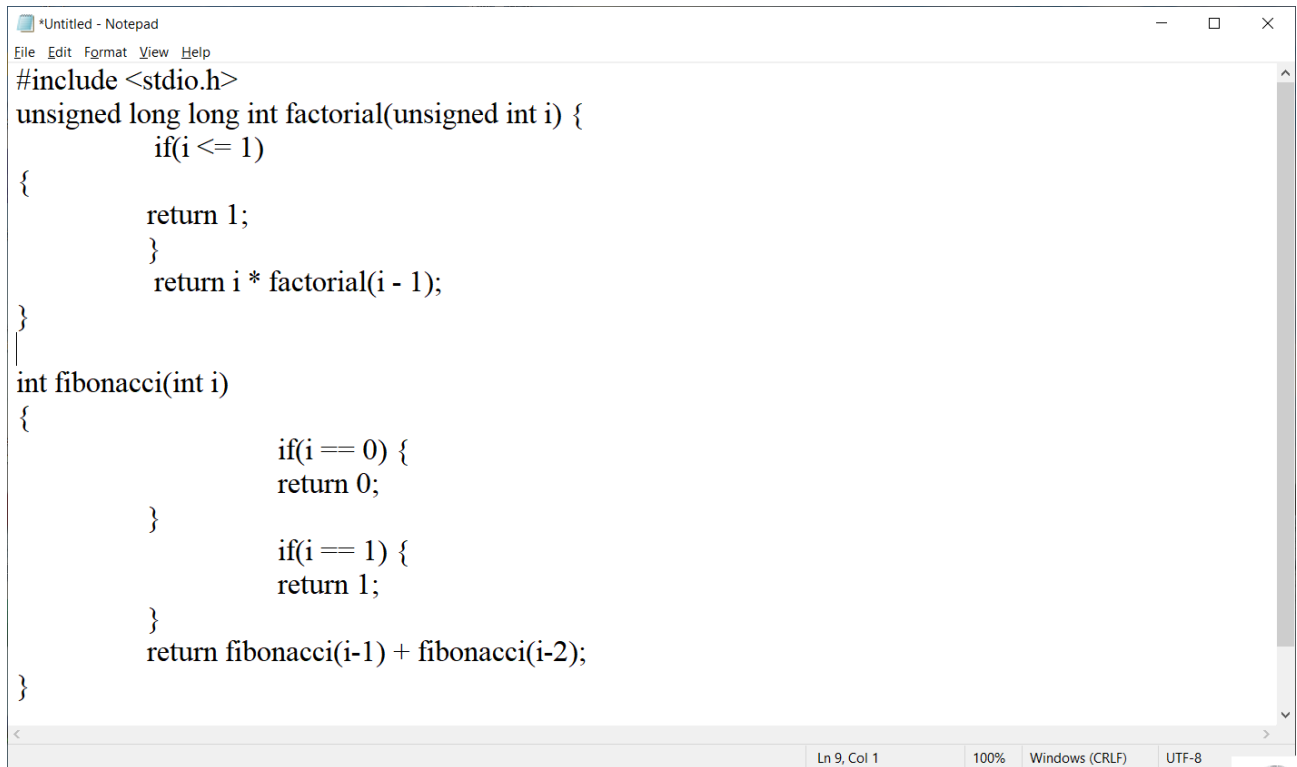
```

#### VI. ការបង្កើត Library File

Function ទាំងឡាយដែលបង្កើតហើយយើងអាចយកវាទៅរក្សាទុកក្នុង Page សរសេរថ្មីមួយហើយ Save as ជា File ដែលមាន Extension .H ដាក់នៅក្នុង Include Folder របស់ Software C យើងនឹងទទួលបាន File Library ហើយយើងអាច Include មកប្រើបានដូច File Library ដែលមានស្រាប់ទុកអោយយើងប្រើដែរ។

## ខាងក្រោមជារូបភាពបង្ហាញពីរបៀបធ្វើ

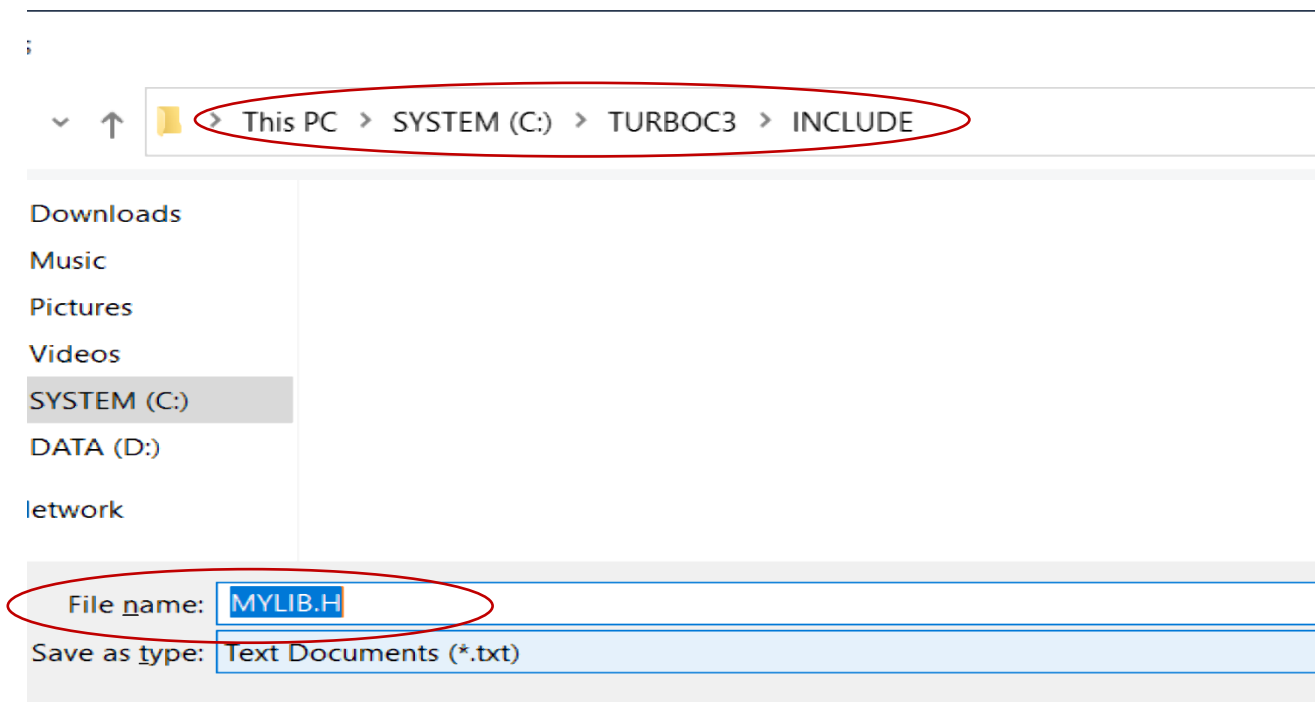
Step1: បង្កើត Function ជាក់ក្នុង Page ថ្មីមួយ



```
*Untitled - Notepad
File Edit Format View Help
#include <stdio.h>
unsigned long long int factorial(unsigned int i) {
    if(i <= 1)
    {
        return 1;
    }
    return i * factorial(i - 1);
}

int fibonacci(int i)
{
    if(i == 0) {
        return 0;
    }
    if(i == 1) {
        return 1;
    }
    return fibonacci(i-1) + fibonacci(i-2);
}
```

Step2: រក្សាទុកជា File .H ក្នុង include Folder របស់ software C

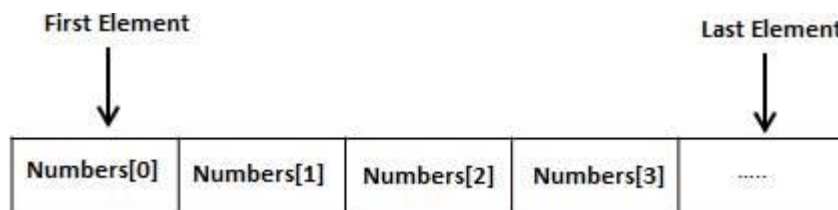


## មេរៀនទី៧

## របៀបប្រើ Array មួយវិមាត្រ

### I. ការប្រើ Array មួយវិមាត្រ

Array គឺជាការផ្គុំនៃការប្រើអថេរដែលមានឈ្មោះដូចគ្នាមានប្រភេទទិន្នន័យដូចគ្នាក្នុងការទុកទិន្នន័យ។ ហើយ Array ប្រើ index នៅក្នុងការគ្រប់គ្រងទិន្នន័យដែលមានតំលៃតូចបំផុតរាប់ចាប់ពីសូន្យឡើងទៅ។ ខាងក្រោមជាទំរង់ការប្រើ Array។



### II. ការបង្កើត Array មួយវិមាត្រ

ដើម្បីប្រកាស Array ក្នុងភាសា C អ្នកសរសេរកម្មវិធីត្រូវបញ្ជាក់ប្រភេទ និងចំនួនធាតុដែលត្រូវការដែលមានទំរង់ដូចខាងក្រោម –

Syntax:

Datatype identifier[size];

Ex1:

float balance[5];

Ex2:

float balance[5]= {1000.0, 2.0, 3.4, 7.0, 50.0};

|         | 0      | 1   | 2   | 3   | 4    |
|---------|--------|-----|-----|-----|------|
| balance | 1000.0 | 2.0 | 3.4 | 7.0 | 50.0 |

### III. ការប្រើ Array មួយវិមាត្រ

Array ប្រើឈ្មោះនិងលេខ index ដាក់នៅចន្លោះ square brackets "[index]" ក្នុងទទួលយកនិងផ្តល់តម្លៃ។ មានទំរង់ដូចឧទាហរណ៍ខាងក្រោម

Ex:

double salary = balance[0];

**ខាងក្រោមជាការបង្ហាញនៃរបៀបប្រើ Array ជានេះ**

```

#include <stdio.h>

int main ( ) {

    int n[ 10 ]; /* n is an array of 10 integers */

    int i,j;

    /* initialize elements of array n to 0 */

    for ( i = 0; i < 10; i++ ) {

        n[ i ] = i + 100; /* set element at location i to i + 100 */

    }

    /* output each array element's value */

    for ( j = 0; j < 10; j++ ) {

        printf( "Element[%d] = %d\n", j, n[j] );

    }

    return 0;

}

```

### **ខាងក្រោមជាលទ្ធផលកម្មវិធីខាងលើ**

Element[0] = 100

Element[1] = 101

Element[2] = 102

Element[3] = 103

Element[4] = 104

Element[5] = 105

Element[6] = 106

Element[7] = 107

Element[8] = 108

Element[9] = 109

### **ខាងក្រោមជាការបង្ហាញនៃរបៀបប្រើ Array ជាអក្សរ**

```
#include <stdio.h>
```

```
int main ( ) {
```

```
    char name[ 5 ][20] = {
```

```
        "dara kok",
```

```
        "bopha kok",
```

```
        "coco kok",
```

```
        "nara kok",
```

```
        "tola kok"
```

```
    };
```

```
    int i,j;
```

```
    /* output each array name value */
```

```
    for ( j = 0; j < 5; j++ ) {
```

```
        printf( "Name[%d] = %s\n", j, name[j] );
```

```
    }
```

```
    return 0;
```

```
}
```

### **ខាងក្រោមជាកូដកម្មវិធីខាងលើ**

```
Name[0] = dara kok
```

Name[1] = bopha kok

Name[2] = coco kok

Name[3] = nara kok

Name[4] = tola kok

#### IV. ការប្រើ Array មួយនិមាត្រជាមួយ Function

ការប្រើ Array ជាមួយ Function តាមរយៈ Parameter យើងអាចប្រើបានបីទំរង់ដូចការបង្ហាញខាងក្រោម

Sample1:

```
void myFunction( int *param ) {  
    .  
    .  
    .  
}
```

Sample2:

```
void myFunction( int param[10] ) {  
    .  
    .  
    .  
}
```

Sample3:

```
void myFunction( int param[ ] ) {  
    .  
    .  
    .  
}
```

#### ខាងក្រោមជាការបង្ហាញនៃការប្រើ

```
double getAverage( int arr[], int size ) {
```

```

        int i;

        double avg;

        double sum = 0;

        for ( i = 0; i < size; ++i ) {

            sum += arr[i];

        }

        avg = sum / size;

        return avg;

    }

#include <stdio.h>

/* function declaration */

double getAverage( int arr[], int size );

int main ( ) {

    /* an int array with 5 elements */

    int balance[5] = {1000, 2, 3, 17, 50};

    double avg;

    /* pass pointer to the array as an argument */

    avg = getAverage( balance, 5 );

    /* output the returned value */

    printf( "Average value is: %f ", avg );

    return 0;

```

}

## ខាងក្រោមជាលទ្ធផលរបស់កម្មវិធីខាងលើ

Average value is: 214.400000

### V. ការប្រើ return Array ពី Function

return Array ពី Function យើងអាចធ្វើតាម return point ។ យើងនឹងនិយាយអំពីការប្រើ Pointer នៅមេរៀនក្រោម។ ខាងក្រោមនឹងបង្ហាញអំពីការប្រើ Function return Array

Sample:

```
int * myFunction( ) {  
    .  
    .  
    .  
}
```

## ខាងក្រោមជាកម្មវិធីបង្ហាញនៃការប្រើ

```
#include <stdio.h>
```

```
/* function to generate and return random numbers */
```

```
int * getRandom( ) {
```

```
    static int r[10];
```

```
    int i;
```

```
    /* set the seed */
```

```
    srand( (unsigned)time( NULL ) );
```

```
    for ( i = 0; i < 10; ++i) {
```

```
        r[i] = rand( );
```

```
        printf( "r[%d] = %d\n", i, r[i] );
```

```
    }
```

```
    return r;
```



```

}

/* main function to call above defined function */

int main ( ) {

    /* a pointer to an int */

    int *p;

    int i;

    p = getRandom( );

    for ( i = 0; i < 10; i++ ) {

        printf( "*(p + %d) : %d\n", i, *(p + i) );

    }

    return 0;

}

```

### **ខាងក្រោមជាលទ្ធផលកម្មវិធីខាងលើ**

```

r[0] = 313959809

r[1] = 1759055877

r[2] = 1113101911

r[3] = 2133832223

r[4] = 2073354073

r[5] = 167288147

r[6] = 1827471542

r[7] = 834791014

r[8] = 1901409888

```

r[9] = 1990469526

\*(p + 0) : 313959809

\*(p + 1) : 1759055877

\*(p + 2) : 1113101911

\*(p + 3) : 2133832223

\*(p + 4) : 2073354073

\*(p + 5) : 167288147

\*(p + 6) : 1827471542

\*(p + 7) : 834791014

\*(p + 8) : 1901409888

\*(p + 9) : 1990469526

## VI. ការប្រើ Search

Search ជាវិធីសាស្ត្រក្នុងស្វែងរកទិន្នន័យដែលមាននៅក្នុង Array ។ ខាងក្រោមជាឧទាហរណ៍នៃការប្រើ

```
/* *
```

```
* C program to search element in array
```

```
*/
```

```
#include <stdio.h>
```

```
#define MAX_SIZE 100 // Maximum array size
```

```
int main( )
```

```
{
```

```
    int arr[MAX_SIZE];
```

```
    int size, i, toSearch, found;
```

```

/* Input size of array */

printf("Enter size of array: ");

scanf("%d", &size);

/* Input elements of array */

printf("Enter elements in array: ");

for( i=0; i<size; i++ )

{

    scanf("%d", &arr[i]);

}

printf("\nEnter element to search: ");

scanf("%d", &toSearch);

/* Assume that element does not exists in array */

found = 0;

for( i=0; i<size; i++ )

{

    /*

    * If element is found in array then raise found flag

    * and terminate from loop.

    */

    if( arr[i] == toSearch )

    {

        found = 1;

```

```

        break;

    }

}

/*

* If element is not found in array

*/

if( found == 1 )

{

    printf( "\n%d is found at position %d", toSearch, i + 1 );

}

else

{

    printf( "\n%d is not found in the array", toSearch );

}

return 0;

}

```

## VII. ការប្រើ Sort

Sort គឺជាការតម្រៀបទិន្នន័យអោយមានលំដាប់ពីរតូចទៅធំឬពីរធំទៅតូច។ ដែរមានវិធីសាស្ត្រពីរក្នុងការអនុវត្ត

- Selection sort
- Bubble sort

### 1. Selection sort

ការតម្រៀប Selection sort គឺជាក្បួនដោះស្រាយមួយដែលត្រូវបានប្រើសម្រាប់ការតម្រៀប។ ក្បួនដោះស្រាយការតម្រៀបនេះ ដោយយកធាតុទាំងអស់របស់ Array ចេញមកហើយស្វែងរកតម្លៃតូចបំផុតក្នុង Array ហើយប្តូរវាជាមួយធាតុទីមួយ ប្រសិនបើវាតូចជាងធាតុទីមួយ។ បន្ទាប់មកវាបន្តទៅធាតុទីពីរ ហើយបន្តរហូតដល់ធាតុទាំងអស់ត្រូវបានតម្រៀប។ ខាងក្រោមជាការឧទាហរណ៍

```

#include <stdio.h>

int main( )
{
    int a[100], n, i, j, position, swap;
    printf( "Enter number of elementsn" );
    scanf( "%d", &n );
    printf( "Enter %d Numbersn", n );
    for ( i = 0; i < n; i++ )
        scanf( "%d", &a[i] );
    for( i = 0; i < n - 1; i++ )
    {
        position=i;
        for( j = i + 1; j < n; j++ )
        {
            if( a[position] > a[j] )
                position=j;
        }
        if( position != i )
        {
            swap=a[i];
            a[i]=a[position];
            a[position]=swap;
        }
    }
    printf( "Sorted Array:n" );
    for( i = 0; i < n; i++ )
        printf( "%dn", a[i] );
    return 0;
}

```

**ខាងក្រោមជាលទ្ធផលកម្មវិធីខាងលើ**

Enter number of elements: 4

Enter 4 Numbers

4

2

7

1

Sorted Array:

1

2

4

7

---

( program exited with code:0 )

## 2. Bubble sort

ការតម្រៀប Bubble sort គឺជាក្បួនដោះស្រាយមួយដែលត្រូវបានប្រើសម្រាប់ការតម្រៀប។ វាជាវិធីសាស្ត្រមួយទៀតខាងក្រោមជាឧទាហរណ៍

```
#include<stdio.h>
```

```
#include<conio.h>
```

```
int main( )
```

```
{
```

```
    int n, i, arr[50], j, temp;
```

```
    printf("Enter the Size ( max. 50): ");
```

```
    scanf("%d",&n);
```

```
    printf("Enter %d Numbers: ",n);
```

```

for( i=0; i<n; i++ )

    scanf( "%d",&arr[i] );

printf( "\nSorting the Array using Bubble Sort Technique..\n" );

for( i=0; i<( n-1 ); i++ )

{

    for( j=0; j<( n-i-1 ); j++ )

    {

        if( arr[j]>arr[j+1] )

        {

            temp = arr[j];

            arr[j] = arr[j+1];

            arr[j+1] = temp;

        }

    }

}

printf( "\nArray Sorted Successfully!\n" );

printf( "\nThe New Array is: \n" );

for( i=0; i<n; i++ ){

    printf( " %d ",arr[i] );

}

printf( "\n~" );

return 0;

```

```
}
```

## ខាងក្រោមជាលទ្ធកម្មវិធីខាងលើ

Enter the Size ( max. 50): 5

Enter 5 Number: 5

1

4

2

3

5

Sorting the Array using Bubble Sort Technique..

Array Sorted Successfully!

The New Array is:

1      2      3      4      5

## VIII. ការប្រើ Delete

Delete ជាវិធីសាស្ត្រលុបចោលធាតុពីការរក្សាទុកក្នុង Array នៅពេលដែលធាតុនោះលេចចំណុចធាតុនេះមានបញ្ហាដោយប្រការណាមួយ។ ខាងក្រោមជាឧទាហរណ៍បង្ហាញពីការលុបចោលធាតុនាមួយចោល

```
/* *
```

```
* C program to delete all duplicate elements from array
```

```
*/
```

```
#include <stdio.h>
```

```
#define MAX_SIZE 100 // Maximum size of the array
```

```
int main( )
```



```

{

int arr[MAX_SIZE]; // Declares an array of size 100

int size;          // Total number of elements in array

int i, j, k;       // Loop control variables

/* Input size of the array */

printf("Enter size of the array : ");

scanf("%d", &size);

/* Input elements in the array */

printf("Enter elements in array : ");

for( i=0; i<size; i++ )

{

    scanf("%d", &arr[i]);

}

/*

    * Find duplicate elements in array

    */

for( i=0; i<size; i++ )

{

    for( j=i+1; j<size; j++ )

    {

        /* If any duplicate found */

        if( arr[i] == arr[j] )

```

```

{
    /* Delete the current duplicate element */

    for( k=j; k < size - 1; k++ )

    {
        arr[k] = arr[k + 1];
    }

    /* Decrement size after removing duplicate element */

    size--;

    /* If shifting of elements occur then don't increment j */

    j--;

}

}

}

/*

* Print array after deleting duplicate elements

*/

printf( "\nArray elements after deleting duplicates : " );

for( i=0; i<size; i++ )

{

    printf( "%d\t", arr[i] );

}

return 0;

```

}

## ខាងក្រោមជាលទ្ធផលកម្មវិធីខាងលើ

Output

Enter size of the array : 10

Enter elements in array : 10 20 10 1 100 10 2 1 5 10

Array elements after deleting duplicates : 10 20 1 100 2 5

### IX. ការប្រើ Edit

Edit ជាវិធីសាស្ត្រកែសម្រួលទិន្នន័យចេញពីការរក្សាទុកក្នុង Array នៅពេលដែលទិន្នន័យនោះមានបញ្ហាដោយប្រការណាមួយ។ខាងក្រោមជាឧទាហរណ៍បង្ហាញ

```
#include<stdio.h>
#include<conio.h>
#define MAX 100
int i,n;
int arr[MAX];
void main( ){
    clrscr( );
    printf( "Please input number of element: " );
    scanf( "%d",&n);
    for( i= 0; i<n; i++ )
    {
        printf( "Element %d : ", i);
        scanf( "%d",&arr[i] );
    }

    printf( "Element in Array before edit is :");

    for( i=0; i<n; i++ )
```

```

    {

        printf( "%d\t",arr[i] );

    }

    printf( "\nEnter number that you want to Edit" );
    scanf( "%d",&x );
    for( i=0; i<n;i++ )
    {
        if( arr[i] == x )
        {
            printf( "Enter new value: " );
            scanf( "%d",&arr[i] );
        }
    }

    printf( "\nElement in Array after edit is :~" );

    for( i=0; i<n; i++ )

    {

        printf( "%d",arr[i] );

    }

    getch( );
}

```

### **ខាងក្រោមជាន់ផ្តល់កម្មវិធីខាងលើ**

Please input number of element: 5

Element[0] : 10

Element[0] : 20

Element[0] : 5

Element[0] : 100

Element[0] : 3

Element in Array before edit is : 10      20      5      100      3

Enter number that you want to Edit: 5

Enter new value: 15

Element in Array after edit is : 10      20      15      100      3

#### X. លំហាត់ការគ្រប់គ្រងព័ត៌មានសិស្សចុះឈ្មោះចូលរៀនសាលារៀនមួយ

សាលារៀនមួយត្រូវការកម្មវិធីគ្រប់គ្រងព័ត៌មានសិស្សរួមមាន លេខកូដ ឈ្មោះ ភេទ ថ្ងៃខែឆ្នាំកំណើត លេខទូរស័ព្ទ ថ្នាក់រៀន ហើយនិងផ្នែកគ្រប់គ្រងរួមមានដូចខាងក្រោម

- ផ្នែកបញ្ចូលព័ត៌មាន
- ផ្នែកមើលព័ត៌មាន
- ផ្នែកស្វែងរកព័ត៌មាន
- ផ្នែកលប់ព័ត៌មាន
- ផ្នែកកែសម្រួលព័ត៌មាន

**សំគាល់:**      ដោយប្រើ Array មួយវិមាត្រ

```
#include<stdio.h>
```

```
#include<conio.h>
```

```
#include<ctype.h>
```

```
#include<string.h>
```

```
#include<dos.h>
```

```
#define MAX 500
```

```
int st_id[MAX];
```

```
char st_name[MAX][30];
```

```
char sex[MAX][7];
```

```
char dob[MAX][15];
```

```
char phone[MAX][15];
```

```

char Class[MAX][5];

int n,i,j,b,x;

char ch;

int main_menu( );

void input( );

void output( );

int search_id( int);

void delete_id( );

void edit_id( );

void main( ){

do{

    clrscr( );

    i=main_menu( );

    switch( i){

    case 0:

        input( );

        break;

    case 1:

        output( );

        break;

    case 2:

        textbackground( BLUE );

```

```

        clrscr( );

        printf( "Enter student id to Search : " );

        scanf( "%d",&x);

        b=search_id(x);

        if( b==0){

            printf( "student id not found" );

            getch( );

        }

        break;

    case 3:

        delete_id( );

        break;

    case 4:

        edit_id( );

        break;

    }

}while( i!=5);

}

void input( ){

    textbackground( BLUE );

    do{

again:

```

```

clrscr( );

printf( "Enter student id: ");

scanf( "%d",&st_id[n]);

b=search_id( st_id[n]);

if( b!=0){

    gotoxy( 1,13);

    printf( "Duplicate id try again: ");

    getch( );

    gotoxy( 1,13);

    clrscr( );

    goto again;

}

printf( "Enter student name: ");

scanf( "%s",&st_name[n]);

printf( "Enter student sex ");

scanf( "%s",&sex[n]);

printf( "Enter student dob: ");

scanf( "%s",&dob[n]);

printf( "Enter student phone: ");

scanf( "%s",&phone[n]);

printf( "Enter student Class: ");

scanf( "%s",&Class[n]);

```



```

        n=n+1;

        printf( "Press Enter continue other exit" );

}while( getch( )==13);

}

void output( ){

textbackground( BLUE );

clrscr( );

if( n<8 )

{

printf( "%10s%15s%10s%15s%15s%10s\n", "ID", "ST_Name", "Sex", "DOB", "Phone", "Class" );

        for( i=0; i<n; i++ )

        {

printf( "%10d%15s%10s%15s%15s%10s\n", st_id[i], st_name[i], sex[i], dob[i], phone[i], Class[i] );

        }

        getch( );

}else

{

    j=0;

    do{

        clrscr( );

printf( "%10s%15s%10s%15s%15s%10s\n", "ID", "ST_Name", "Sex", "DOB", "Phone", "Class" );

        for( i=j; i<j+8; i++ )

        {

```

```

printf( "%10d%15s%10s%15s%15%10s\n",st_id[i],st_name[i],sex[i],dob[i],phone[i],Class[i] );

    }

    ch=getch( );

    if( tolower( ch )=='a' )

    {

        j--; if( j<0 ) j++;

    }

    if( tolower( ch )=='z' )

    {

        j++; if( j+8>n ) j--;

    }

}while( ch!=27 );

}

}

int search_id( int x ){

    b=0;

    for( i=0;i<n;i++ )

    {

        if( st_id[i]==x ){ b=1;

            printf( "\nStudent id:  %d\n",st_id[i] );

            printf( "Student name: %s\n",st_name[i] );

            printf( "Student sex:  %s\n",sex[i] );

```

```

        printf( "Student DoB: %s\n",dob[i] );

        printf( "Student Phone:%s\n",phone[i] );

        printf( "Student Class:%s\n",Class[i] );

    }

}

if( b!=0 ) getch( );

return b;

}

void delete_id( ){

    textbackground( BLUE );

    clrscr( );

    printf( "Enter student id to delete: " );scanf( "%d",&x );

    b=0;

    for( i=0;i<n;i++ )

    {

        if( st_id[i]==x )

        {

            b=1;

            for( j=i;j<n-1;j++ )

            {

                st_id[j]=st_id[j+1];

                strcpy( st_name[j],st_name[j+1] );

```

```

        strcpy( sex[j],sex[j+1] );

        strcpy( dob[j],dob[j+1] );

        strcpy( phone[j],phone[j+1] );

        strcpy( Class[j],Class[j+1] );

    }

    n=n-1;

}

}

if( b==0 )

    printf( "Delete not completed" );

else

    printf( "Delete completed" );

    getch( );

}

void edit_id( ){

    textbackground( BLUE );

    clrscr( );

    printf( "Enter student id to Edit: " );scanf( "%d",&x);

    b=0;

    for( i=0;i<n;i++ )

    {

```

```

        if( st_id[i]==x)
        {
            b=1;

            printf( "Enter student name: ");

            scanf( "%s",&st_name[i] );

            printf( "Enter student sex ");

            scanf( "%s",&sex[i] );

            printf( "Enter student dob: ");

            scanf( "%s",&dob[i] );

            printf( "Enter student phone: ");

            scanf( "%s",&phone[i] );

            printf( "Enter student Class: ");

            scanf( "%s",&Class[i] );

        }
    }

    if( b==0){

        printf( "Edit not completed" );

    }

    Else{

        printf( "Edit Update completed" );

    }

    getch( );

```

```

}

int main_menu( ){

do{

    textbackground( BLUE );

    clrscr( );

    textcolor( WHITE );

    textbackground( GREEN );

    gotoxy( 1,10+i );

    cprintf( "%30c",' ');

    gotoxy( 1,10 );

    printf( "Register information student" );

    gotoxy( 1,11 );

    printf( "Show information student" );

    gotoxy( 1,12 );

    printf( "Search information student" );

    gotoxy( 1,13 );

    printf( "Delete information student" );

    gotoxy( 1,14 );

    printf( "Edit information student" );

    gotoxy( 1,15 );

    printf( "Exit program\n" );

    gotoxy( 1,25 );

```

```

printf( "Use key to Scroll 'A' move 'UP' or 'Z' move 'Down'");

printf( " Enter='Select'");

ch=getch( );

switch( tolower( ch ) ){

case 'a':

    i=i-1;

    if( i<0 ) i=5;

    break;

case 'z':

    i=i+1;

    if( i>5 ) i=0;

}

}while( ch!=27 && ch!=13);

if( ch==27 )

{

    i=5;

}

return i;

}

```

## មេរៀនទី៨

## របៀបប្រើ Array ប្រើនវិមាត្រ

### I. ការប្រើ Array ប្រើនវិមាត្រ

Array ប្រើនវិមាត្រគឺជា Array ដែលមានធាតុរបស់ជា Array បន្តទៀតហើយមិនមានកំណត់ទៅខាងក្រោមជាទំរង់ការប្រើ Array។

Syntax:

```
Datatype identifier[size][size]...[size_n];
```

Ex:

```
int a[5][3][2];
```

### II. ការបង្កើត Array ពីរវិមាត្រ

ទម្រង់សាមញ្ញបំផុតនៃ Array ប្រើនវិមាត្រគឺ Array ពីរវិមាត្រ។ Array ពីរវិមាត្រជា Array មួយវិមាត្រដែលមានធាតុរបស់វាជា Array មួយវិមាត្របន្តទៀត ។ ដើម្បីប្រកាស Array ចំនួនគត់ពីរវិមាត្រមានទំហំ [x][y] អ្នកនឹងទទួលបានទំរង់ដូចខាងក្រោម –

Syntax:

```
Datatype identifier[x][y];
```

Ex:

```
int a[3][4];
```

ខាងក្រោមជាការបង្ហាញទំរង់នៃការប្រើធាតុរបស់ Array ពីរវិមាត្រ

|       | Column 0    | Column 1    | Column 2    | Column 3    |
|-------|-------------|-------------|-------------|-------------|
| Row 0 | a[ 0 ][ 0 ] | a[ 0 ][ 1 ] | a[ 0 ][ 2 ] | a[ 0 ][ 3 ] |
| Row 1 | a[ 1 ][ 0 ] | a[ 1 ][ 1 ] | a[ 1 ][ 2 ] | a[ 1 ][ 3 ] |
| Row 2 | a[ 2 ][ 0 ] | a[ 2 ][ 1 ] | a[ 2 ][ 2 ] | a[ 2 ][ 3 ] |

### ខាងក្រោមជាការបង្ហាញនៃការផ្តល់តំលៃដល់អោយ Array ពីរវិមាត្រ

```
int a[3][4] = {  
    {0, 1, 2, 3} , /* initializers for row indexed by 0 */  
    {4, 5, 6, 7} , /* initializers for row indexed by 1 */  
    {8, 9, 10, 11} /* initializers for row indexed by 2 */  
};
```



### III. ការប្រើ Array ពីរវិមាត្រ

ការប្រើ Array ពីរវិមាត្រនៃធាតុណាមួយត្រូវប្រើតម្លៃនៃធាតុជួរដេកនិងជួរឈរ។ ខាងក្រោមជាការបង្ហាញនៃការប្រើ Array ពីរវិមាត្រ

Ex:

```
int val = a[2][3];
```

### ខាងក្រោមជាឧទាហរណ៍កម្មវិធីនៃការប្រើ Array ពីរវិមាត្រ

```
#include <stdio.h>

int main ( ) {
    /* an array with 5 rows and 2 columns */
    int a[5][2] = { {0,0}, {1,2}, {2,4}, {3,6},{4,8}};
    int i, j;
    /* output each array element's value */
    for ( i = 0; i < 5; i++ ) {
        for ( j = 0; j < 2; j++ ) {
            printf( "a[%d][%d] = %d\n", i,j, a[i][j] );
        }
    }
    return 0;
}
```

### ខាងក្រោមជាលទ្ធផលនៃកម្មវិធីខាងលើ

```
a[0][0]: 0
a[0][1]: 0
a[1][0]: 1
a[1][1]: 2
a[2][0]: 2
a[2][1]: 4
a[3][0]: 3
a[3][1]: 6
a[4][0]: 4
a[4][1]: 8
```

#### IV. ការប្រើ Array ពីវិមាត្រជាអក្សរ

Array ពីវិមាត្រជាអក្សរ ឬ array មួយវិមាត្រជា String ជាការផ្ទុកទិន្នន័យអក្សរជាច្រើននៅក្នុង Array ។ មានទំរង់នៃការប្រើដូចខាងក្រោម

Syntax:

```
char a[size][size];
```

Ex:

```
char Name[3][20]={ "dara kok","Bopha kok","nana kok"};
```

#### ខាងក្រោមជាឧទាហរណ៍កម្មវិធីនៃការប្រើ Array ពីវិមាត្រជាអក្សរ

```
#include <stdio.h>
#include<conio.h>
int main ( ) {
    int a[5][20] = {"dara kok","bopha kok","nana kok","tola kok","thyda kok"};
    int i, j;
    /* output each array element's value */
    clrscr( );
    for ( i = 0; i < 5; i++ ) {
        printf( "Name[%d]= ",i);
        for ( j = 0; j < 2; j++ ) {
            printf( "%c", a[i][j] );
        }
        printf( "\n" );
    }
    return 0;
}
```

#### ខាងក្រោមជាលទ្ធផលនៃកម្មវិធីខាងលើ

Name[0]: dara kok

Name[1]: bopha kok

Name[2]: nana kok

Name[3]: tola kok

Name[4]: thyda kok

## V. ការប្រើ Search

Search ជាវិធីសាស្ត្រក្នុងស្វែងរកទិន្នន័យដែលមាននៅក្នុង Array ។ ខាងក្រោមជាឧទាហរណ៍នៃការប្រើ Array ពីរវិមាត្រ

```
#include<stdio.h>
#include<conio.h>
#include<string.h>
#define MAX 100
char name[MAX][20];
int i,b,n;
char x[20];
void main( )
{
    clrscr( );
    printf( "Enter number of person: " );
    scanf( "%d", & n );
    for( i = 0; i<n ; i++ )
    {
        printf( "Enter person name: " );
        scanf( "%s",&name[i] );
    }
    printf( "Enter person name to search : " );
    scanf( "%s",&x );
    b=0;
    for( i=0; i<n; i++ ){
        if( strcmp( name[i],x)==0 )
        {
            b=1;
            printf( "Name: %s found in array index %d",name[i], i );
        }
    }
    if( b==0 )
```

```

        printf( "person name not found" );

        getch( );
    }

```

### **ខាងក្រោមជារង្វង់លទ្ធផលទី១របស់កម្មវិធីខាងលើ**

Enter number of person: 5  
 Enter person name: Narith  
 Enter person name: Thyda  
 Enter person name: Vuthy  
 Enter person name: Daro  
 Enter person name: Kunthea  
 Enter person name to search : Thyda  
 Name: Thyda found in array index 1

### **ខាងក្រោមជារង្វង់លទ្ធផលទី២របស់កម្មវិធីខាងលើ**

Enter number of person: 5  
 Enter person name: Narith  
 Enter person name: Thyda  
 Enter person name: Vuthy  
 Enter person name: Daro  
 Enter person name: Kunthea  
 Enter person name to search : Navy  
 Person name not found

## **VI. ការប្រើ Sort**

Sort គឺជាការតម្រៀបទិន្នន័យអោយមានលំដាប់ពីរតូចទៅធំឬពីធំទៅតូច។ ខាងក្រោមជាវិធីសាស្ត្រតម្រៀប Array ពីរវិមាត្រតាមលំដាប់

```

#include<stdio.h>
#include<conio.h>
#include<string.h>
#define MAX 100
char name[MAX][20];
int i,j,n;

```

```

char x[20];
void main( )
{
    clrscr( );
    printf( "Enter number of person: " );
    scanf( "%d", & n );
    for( i = 0; i<n ; i++ )
    {
        printf( "Enter person name: " );
        scanf( "%s",&name[i] );
    }
    printf( "Information before sort\n " );
    for( i = 0; i<n ; i++ )
    {
        printf( "Person name: %s\n",name[i] );
    }
    for( i=0; i<n; i++ )
    {
        for( j=i+1; j<n; j++ )
        {
            if( strcmp( name[i],name[j] )>0 )
            {
                strcpy( x,name[i] );
                strcpy( name[i],name[j] );
                strcpy( name[j],x );
            }
        }
    }
    printf( "Information after sort\n " );
    for( i = 0; i<n ; i++ )
    {
        printf( "Person name: %s\n",name[i] );
    }
}

```

```
}
getch( );
}
```

### **ខាងក្រោមជាលទ្ធផលរបស់កម្មវិធីខាងលើ**

```
Enter number of person: 5
Enter person name: Narith
Enter person name: Thyda
Enter person name: Vuthy
Enter person name: Daro
Enter person name: Kunthea
Information before sort
Person name: Narith
Person name: Thyda
Person name: Vuthy
Person name: Daro
Person name: Kunthea
Information After sort
Person name: Daro
Person name: Kunthea
Person name: Narith
Person name: Thyda
Person name: Vuthy
```

## **VII. ការប្រើ Delete**

Delete ជាវិធីសាស្ត្រលុបទិន្នន័យចេញពីការរក្សាទុកក្នុង Array នៅពេលដែលទិន្នន័យនោះលែងចាំបាច់។ ទិន្នន័យមានបញ្ហាដោយប្រការណាមួយ។ ខាងក្រោមជាឧទាហរណ៍បង្ហាញពីការលុបទិន្នន័យណាមួយក្នុង Array ពីរវិមាត្រជាអក្សរ

```
#include<stdio.h>
#include<conio.h>
#include<string.h>
#define MAX 100
```

```

char name[MAX][20];
int i,j,n,b;
char x[20];
void main( )
{
    clrscr( );
    printf( "Enter number of person: " );
    scanf( "%d", & n );
    for( i = 0; i<n ; i++ )
    {
        printf( "Enter person name: " );
        scanf( "%s",&name[i] );
    }

    printf( "Enter person name to delete : " );
    scanf( "%s",&x );
    b=0;
    for( i=0; i<n; i++ ){
        if( strcmp( name[i],x)==0 )
        {
            b=1;
            for( j=i; j<n; j++ )
            {
                strcpy( name[j],name[j+1] );
            }

            n--;
        }
    }
    if( b==0 )

        printf( "Delete not completed" );
}

```

```

else

printf( "Delete completed" );

getch( );
}

```

### **ខាងក្រោមជាលទ្ធផលទី១របស់កម្មវិធីខាងលើ**

```

Enter number of person: 5
Enter person name: Narith
Enter person name: Thyda
Enter person name: Vuthy
Enter person name: Daro
Enter person name: Kunthea
Enter person name to delete: Thyda
Delete completed

```

### **ខាងក្រោមជាលទ្ធផលទី២របស់កម្មវិធីខាងលើ**

```

Enter number of person: 5
Enter person name: Narith
Enter person name: Thyda
Enter person name: Vuthy
Enter person name: Daro
Enter person name: Kunthea
Enter person name to delete: Koko
Delete not completed

```

## **VIII. ការប្រើ Edit**

Edit ជាវិធីសាស្ត្រកែសម្រួលទិន្នន័យចេញពីការរក្សាទុកក្នុង Array នៅពេលដែលទិន្នន័យនោះមានបញ្ហាដោយប្រការណាមួយ។ ខាងក្រោមជាឧទាហរណ៍បង្ហាញពីការកែសម្រួលឈ្មោះពីរក្នុង Array ពីរវិមាត្រជាអក្សរ

```

#include<stdio.h>
#include<conio.h>
#include<string.h>
#define MAX 100

```



```

char name[MAX][20];
int i,j,n,b;
char x[20];
void main( )
{
    clrscr( );
    printf( "Enter number of person: " );
    scanf( "%d", & n );
    for( i = 0; i<n ; i++ )
    {
        printf( "Enter person name: " );
        scanf( "%s",&name[i] );
    }

    printf( "Enter person name to Edit : " );
    scanf( "%s",&x );
    b=0;
    for( i=0; i<n; i++ ){
        if( strcmp( name[i],x )==0 )
        {
            b=1;
            printf( "Enter new person name: " );
            scanf( "%s",&name[i] );
        }
    }
    if( b==0 )
        printf( "Edit not completed" );

    else
        printf( "Edit completed" );

    getch( );
}

```

}

**ខាងក្រោមជាលទ្ធផលទី១របស់កម្មវិធីខាងលើ**

Enter number of person: 5

Enter person name: Narith

Enter person name: Thyda

Enter person name: Vuthy

Enter person name: Daro

Enter person name: Kunthea

Enter person name to Edit: Thyda

Enter new person name: Thyrith

Edit completed

**ខាងក្រោមជាលទ្ធផលទី២របស់កម្មវិធីខាងលើ**

Enter number of person: 5

Enter person name: Narith

Enter person name: Thyda

Enter person name: Vuthy

Enter person name: Daro

Enter person name: Kunthea

Enter person name to Edit: Koko

Edit not completed

**IX. លំហាត់ការគ្រប់គ្រងព័ត៌មានសិស្សចុះឈ្មោះចូលរៀនសាលារៀនមួយ ( Update )**

សាលារៀនមួយត្រូវការកម្មវិធីគ្រប់គ្រងព័ត៌មានសិស្សរួមមាន លេខកូដ ឈ្មោះ ភេទ ថ្ងៃខែឆ្នាំកំណើត លេខទូរស័ព្ទ ថ្នាក់រៀន ហើយនិងផ្នែកគ្រប់គ្រងរួមមានដូចខាងក្រោម

- ផ្នែកបញ្ចូលព័ត៌មាន
- ផ្នែកមើលព័ត៌មាន
- ផ្នែកស្វែងរកព័ត៌មាន
- ផ្នែកលប់ព័ត៌មាន
- ផ្នែកកែសម្រួលព័ត៌មាន

**សំគាល់:** ដោយប្រើ Array ពីរវិមាត្រ

```
#include<stdio.h>

#include<conio.h>

#include<ctype.h>

#include<string.h>

#include<dos.h>

#define MAX 100

char stu[MAX][6][30];

/* Update from Array one

int st_id[MAX];

char st_name[MAX][30];

char sex[MAX][7];

char dob[MAX][15];

char phone[MAX][15];

char Class[MAX][5];

*/

int n,i,j,b;

char ch;

char x[20];

int main_menu( );

void input( );

void output( );
```

```

int search_id( char * );

void delete_id( );

void edit_id( );

void main( ){

do{

    clrscr( );

    i=main_menu( );

    switch(i){

    case 0:

        input( );

        break;

    case 1:

        output( );

        break;

    case 2:

        textbackground( BLUE );

        clrscr( );

        printf( "Enter student id to Search : " );

        scanf( "%s",&x );

        b=search_id( x );

        if( b==0 ){

            printf( "student id not found" );

```

```

        getch( );

    }

    break;

case 3:

    delete_id( );

    break;

case 4:

    edit_id( );

    break;

}

}while( i!=5);

}

void input( ){

    textbackground( BLUE );

do{

again:

    clrscr( );

    printf( "Enter student id: " );

    scanf( "%s",&stu[n][0] );

    b=search_id( stu[n][0] );

    if( b!=0 ){

        gotoxy( 1,13 );

```

```

        printf( "Duplicate id try again: ");

        getch( );

        gotoxy( 1,13);

        clrscr( );

        goto again;

    }

    printf( "Enter student name: ");

    scanf( "%s",&stu[n][1] );

    printf( "Enter student sex ");

    scanf( "%s",&stu[n][2] );

    printf( "Enter student dob: ");

    scanf( "%s",&stu[n][3] );

    printf( "Enter student phone: ");

    scanf( "%s",&stu[n][4] );

    printf( "Enter student Class: ");

    scanf( "%s",&stu[n][5] );

    n=n+1;

    printf( "Press Enter continue other exit" );

}while( getch( )!=13);

}

void output( ){

textbackground( BLUE );

```

```

clrscr( );

if( n<8 )

{

printf( "%10s%15s%10s%15s%15s%10s\n", "ID", "ST_Name", "Sex", "DOB", "Phone", "Class" );

    for( i=0; i<n; i++ )

    {

printf( "%10s%15s%10s%15s%15s%10s\n", stu[i][0], stu[i][1], stu[i][2], stu[i][3], stu[i][4], stu[i][5] );

    }

    getch( );

}else

{
    j=0;

    do{

        clrscr( );

printf( "%10s%15s%10s%15s%15s%10s\n", "ID", "ST_Name", "Sex", "DOB", "Phone", "Class" );

        for( i=j; i<j+8; i++ )

        {

printf( "%10s%15s%10s%15s%15s%10s\n", stu[i][0], stu[i][1], stu[i][2], stu[i][3], stu[i][4], stu[i][5] );

        }

        ch=getch( );

        if( tolower( ch)=='a' )

        {

            j--; if( j<0 ) j++;

```

```

    }

    if( tolower( ch )=='z' )

    {

        j++; if( j+8>n ) j--;

    }

}while( ch !=27 );

}

}

int search_id( char *x){

    b=0;

    for( i=0;i<n;i++)

    {

        if( strcmp( stu[i][0],x)==0 ){ b=1;

            printf( "\nStudent id: %s\n",stu[i][0] );

            printf( "Student name: %s\n",stu[i][1] );

            printf( "Student sex: %s\n",stu[i][2] );

            printf( "Student DoB: %s\n",stu[i][3] );

            printf( "Student Phone:%s\n",stu[i][4] );

            printf( "Student Class:%s\n",stu[i][5] );

        }

    }

    if( b!=0 ) getch( );

```



```

        return b;
    }

void delete_id( ){

    int k;

    textbackground( BLUE );

    clrscr( );

    printf( "Enter student id to delete: " );scanf( "%s",&x );

    b=0;

    for( i=0;i<n;i++ )

    {

        if( strcmp( stu[i][0],x)==0 )

        {

            b=1;

            for( j=i;j<n-1;j++ )

            {

                for( k=0;k<6;k++ )

                    strcpy( stu[j][k],stu[j+1][k] );

            }

            n=n-1;

        }

    }

    if( b==0 )

```

```

        printf( "Delete not completed" );

    else

        printf( "Delete completed" );

    getch( );

}

void edit_id( ){

    textbackground( BLUE );

    clrscr( );

    printf( "Enter student id to Edit: " );scanf( "%s",&x );

    b=0;

    for( i=0;i<n;i++ )

    {

        if( strcmp( stu[i][0],x)==0 )

        {

            b=1;

            printf( "Enter student name: " );

            scanf( "%s",&stu[i][1] );

            printf( "Enter student sex " );

            scanf( "%s",&stu[i][2] );

            printf( "Enter student dob: " );

            scanf( "%s",&stu[i][3] );

            printf( "Enter student phone: " );

```

```

        scanf( "%s",&stu[i][4] );

        printf( "Enter student Class: " );

        scanf( "%s",&stu[i][5] );

    }

}

if( b==0 )

{

    printf( "Edit not completed" );

}else

{

    printf( "Edit Update completed" );

}

getch( );

}

int main_menu( )

{

do{

    textbackground( BLUE );

    clrscr( );

    textcolor( WHITE );

    textbackground( GREEN );

    gotoxy( 1,10+i );

```

```

cprintf( "%30c", ' ');

gotoxy( 1,10 );

printf( "Register information student" );

gotoxy( 1,11 );

printf( "Show information student" );

gotoxy( 1,12 );

printf( "Search information student" );

gotoxy( 1,13 );

printf( "Delete information student" );

gotoxy( 1,14 );

printf( "Edit information student" );

gotoxy( 1,15 );

printf( "Exit program\n" );

gotoxy( 1,25 );

printf( "Use key to Scroll 'A' move 'UP' or 'Z' move 'Down'" );

printf( " Enter='Select'" );

ch=getch( );

switch( tolower( ch ) ){

case 'a':

    i=i-1;

    if( i<0 ) i=5;

    break;

```

```
        case 'z':  
            i=i+1;  
            if( i>5 ) i=0;  
        }  
    }while( ch!=27 && ch!=13 );  
    if( ch==27 ) i=5;  
    return i;  
}
```

## មេរៀនទី៩

## របៀបប្រើ String

### I. និយមន័យ String

String គឺជា Array មួយវិមាត្រនៃតួអក្សរដែលត្រូវបានបញ្ចប់ដោយតួអក្សរទទេ '\0' ។ ខាងក្រោមជាការបង្ហាញទំរង់ទូទៅរបស់ String។

Syntax:

```
char name[size];    // String is array of character
```

Ex:

```
char name[size];
```

### II. របៀបប្រើ String

String ការផ្តល់តម្លៃអាចប្រើដោយធាតុមួយៗចាប់ពីសូន្យ ឬ ក៏ប្រើឈ្មោះក្នុងការទទួលតម្លៃ។ ខាងក្រោមជាការបង្ហាញនៃការប្រើ។

Ex:

```
char greeting[6] = {'H', 'e', 'l', 'l', 'o', '\0'};
```

or

```
char greeting[] = "Hello";
```

| Index    | 0       | 1       | 2       | 3       | 4       | 5       |
|----------|---------|---------|---------|---------|---------|---------|
| Variable | H       | e       | l       | l       | o       | \0      |
| Address  | 0x23451 | 0x23452 | 0x23453 | 0x23454 | 0x23455 | 0x23456 |

### ខាងក្រោមជាកម្មវិធីនៃការប្រើ String

```
#include <stdio.h>
```

```
int main ( ) {
```

```
    char greeting[6] = {'H', 'e', 'l', 'l', 'o', '\0'};
```

```
    printf( "Greeting message: %s\n", greeting );
```

```
    return 0;
```

}

## ខាងក្រោមជាលទ្ធផលនៃកម្មវិធីខាងលើ

Greeting message: Hello

### III. ការប្រើប្រាស់ប្រភេទទិន្នន័យ String

String នៅក្នុងភាសា C មានមុខងារជាច្រើនដែលត្រូវបានប្រើដោយឃ្លាបញ្ជាខាងក្រោម។

| Sr.No. | ប្រភេទទិន្នន័យ និង ការពិពណ៌នា                                                                                                                            |
|--------|----------------------------------------------------------------------------------------------------------------------------------------------------------|
| 1      | <code>gets( s1 );</code><br>បញ្ចូលទិន្នន័យជាអក្សរច្រើន (អាចដកឃ្លាបាន)                                                                                    |
| 2      | <code>strcpy( s1, s2 );</code><br>ចម្លង string s2 ទៅ string s1.                                                                                          |
| 3      | <code>strncpy( s1, s2,n );</code><br>ចម្លង string s2 ទៅ string s1 មានចំនួន n កំនត់.                                                                      |
| 4      | <code>strcat( s1, s2 );</code><br>បន្ថែម string s2 អោយទៅ string s1.                                                                                      |
| 5      | <code>strncat( s1, s2 );</code><br>បន្ថែម string s2 អោយទៅ string s1 មានចំនួន n កំនត់.                                                                    |
| 6      | <code>strlen( s1 );</code><br>ប្រាប់អំពីចំនួនអក្សរ string s1.                                                                                            |
| 7      | <code>strcmp( s1, s2 );</code><br>ប្រាប់ 0 ប្រសិន បើ s1 និង s2 ជាទិន្នន័យដូចគ្នា។ ប្រាប់តូចជាង 0 ប្រសិនបើ $s1 < s2$ ។ ប្រាប់ធំជាង 0 ប្រសិនបើ $s1 > s2$ . |

|    |                                                                                                                                                            |
|----|------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 8  | <b>strncmp( s1, s2 );</b><br>ប្រាប់ 0 ប្រសិន s1 និង s2 ជាទិន្នន័យដូចគ្នា។ ប្រាប់តូចជាង 0 ប្រសិនបើ s1<s2<br>។ប្រាប់ធំជាង 0 ប្រសិនបើ s1>s2 ។មានចំនួន n កំនត់ |
| 9  | <b>strchr( s1, ch );</b><br>ប្រាប់នៅទីតាំងរបស់តួអក្សរ ch ដំបូងដែលមាននៅក្នុង String s1                                                                      |
| 10 | <b>strstr( s1, s2 );</b><br>ប្រាប់នៅទីតាំងរបស់ String S2 ដំបូងដែលមាននៅក្នុង String s1                                                                      |

### ខាងក្រោមជាកម្មវិធីបង្ហាញពីការប្រើប្រាស់បញ្ជាខាងលើ

```
#include <stdio.h>

#include <string.h>

int main ( ) {

    char str1[12] = "Hello";

    char str2[12] = "World";

    char str3[12];

    int len ;

    /* copy str1 into str3 */

    strcpy( str3, str1 );

    printf( "strcpy( str3, str1 ) : %s\n", str3 );

    /* concatenates str1 and str2 */

    strcat( str1, str2 );
```



```

printf("strcat( str1, str2): %s\n", str1 );

/* total length of str1 after concatenation */

len = strlen( str1 );

printf("strlen( str1 ) : %d\n", len );

printf("Enter new string: "); gets( st3 );

if( strcmp( st3,st2 )==0 )

    printf("String 2 and String 3 is the same");

printf("In String 1 have character W in index: %d", strchr( st1,'W' ) );

printf("In String 1 have word 'ell' in index: %d",strstr( st1,"ell" ) );

return 0;

}

```

### **ខាងក្រោមជាលទ្ធផលកម្មវិធីខាងលើ**

```

strcpy( str3, str1 ) : Hello
strcat( str1, str2): HelloWorld
strlen( str1 ) : 10
Enter new string: World
String 2 and String 3 is the same
In String 1 have character W in index: 5
In String 1 have word 'ell' in index: 1

```

#### **IV. ការប្រើប្រតិបត្តិលក្ខខណ្ឌ String**

String ជាប្រភេទទិន្នន័យដែលអ្នកប្រើអាចផ្តល់ទិន្នន័យជាប្រភេទណាមួយក៏បានដូច្នេះអ្នកសរសេរកម្មវិធីត្រូវដឹងអំពីការប្រតិបត្តិលក្ខខណ្ឌក្នុងការផ្តល់ទិន្នន័យ String ។ខាងក្រោមជាការបង្ហាញអំពីការ ប្រតិបត្តិលក្ខខណ្ឌក្នុងការផ្តល់ទិន្នន័យ String ត្រូវតែជាតម្លៃ។

```
#include<stdio.h>
```

```

#include<conio.h>
#define MAX 255
char ch;
char st[MAX]="";
int i=0;
void inputs( char * );
void main( ){
    clrscr( );
    printf( "Please Enter your Phone number: " );inputs( st );
    printf( "Your phone is %s",st );
    getch( );
}
void inputs( char *st ){
    do{
        ch=getch( );
        if( ch>='0' && ch<='9' || ch==8 )
        {
            if( ch!=8 )
            {
                st[i]=ch; i++;
                printf( "%c",ch );
            }
            else
            {
                if( i>0 )
                {
                    i--;
                    st[i]='\0';
                    printf( "%c",ch );
                    printf( " " );
                    gotoxy( wherex( )-1,wherey( ) );

```

```

    }
}
}
}while( ch!=13);
}

```

### ខាងក្រោមជាលទ្ធផលកម្មវិធីខាងលើ

Please Enter your Phone number: 0979932456

Your phone is: 0979932456

//User must use key number or Backspace and other not access

### V. ការបំប្លែង String ទៅជាលេខ

String ជាប្រភេទទិន្នន័យដែលអ្នកប្រើអាចផ្តល់ទិន្នន័យជាប្រភេទណាមួយក៏បានបើអ្នកចង់បាន String ផ្លាស់ប្តូរ ប្រភេទទៅជាប្រភេទលេខខាងក្រោមជាការបង្ហាញនៃការប្រើ

Syntax:

```
atoi( string ); // convert string to integer
```

```
atof( string ); // convert string to float
```

Ex:

```
#include <stdlib.h>
```

```
#include <stdio.h>
```

```
int main( void )
```

```
{
```

```
    int n;
```

```
    char *str = "12345.67";
```

```
    n = atoi( str );
```

```
    printf( "string = %s integer = %d\n", str, n );
```

```
    return 0;
```

```
}
```

### ខាងក្រោមជាលទ្ធផលកម្មវិធីខាងលើ

String = 12345.67 integer = 12345

### ខាងក្រោមជាឧទាហរណ៍នៃការបំប្លែង String ទៅជាតម្លៃទសភាគ

```
#include <stdlib.h>
#include <stdio.h>
int main( void )
{
    float f;
    char *str = "12345.67";
    f = atof( str );
    printf( "string = %s float = %f\n", str, f );
    return 0;
}
```

**ខាងក្រោមជាលទ្ធកម្មវិធីខាងលើ**

String = 12345.67 float = 12345.67

## VI. ការបំប្លែងលេខទៅជា String

String ជាប្រភេទទិន្នន័យដែលអ្នកប្រើអាចផ្តល់ទិន្នន័យជាប្រភេទណាមួយក៏បានបើអ្នកចង់បានទិន្នន័យជាលេខបំប្លែងប្រភេទទៅជា String ខាងក្រោមជាការបង្ហាញនៃការប្រើ

Syntax:

```
itoa( number, string, radix );
```

- Number : ជាតម្លៃលេខណាមួយដែលត្រូវបំប្លែងទៅជា String
- String : ជាអថេរអក្សរច្រើនណាមួយចាំទទួល
- Radix : ជាលេខគិតដោយប្រើប្រព័ន្ធណាមួយ ( Ex: 2,8,10,16 )

Syntax:

```
gcvt( Number, Sig, str );
```

- Number : ជាលេខទសភាគដែលអ្នកចង់បំប្លែង
- Sig : ជាចំនួនតួលេខដែលត្រូវការ
- String : ជាអថេរអក្សរច្រើនចាំទទួល

```
#include <stdlib.h>
```

```
#include <stdio.h>
```

```

int main( void )

{

    int number = 12345;

    char string[25];

    itoa( number, string, 10 );

    printf( "integer = %d string = %s\n", number, string );

    return 0;

}

```

### **ខាងក្រោមជាលទ្ធផលកម្មវិធីខាងលើ**

integer = 12345 string = 12345

### **ខាងក្រោមជាឧទាហរណ៍នៃការបំប្លែងតម្លៃទសភាគអោយទៅជា String**

```

#include <stdlib.h>

#include <stdio.h>

#include<conio.h>

int main( void )

{

    char str[25];

    double num;

    int sig = 5; /* significant digits */

    /* a regular number */

    num = 9.876;

    gcvt( num, sig, str );

```

```

clrscr( );

printf( "string = %s\n", str );

/* a negative number */

num = -123.4567;

gcvtf( num, sig, str );

printf( "string = %s\n", str );

/* scientific notation */

num = 0.678e5;

gcvtf( num, sig, str );

printf( "string = %s\n", str );

getch( );

return( 0 );

}

```

### **ខាងក្រោមជាលក្ខណៈកម្មវិធីខាងលើ**

```

string = 9.876

string = -123.4567

string = 0.678e5

```

### **VII. លំហាត់ការគ្រប់គ្រងព័ត៌មានសិស្សចុះឈ្មោះចូលរៀនសាលារៀនមួយ ( Update )**

សាលារៀនមួយត្រូវការកម្មវិធីគ្រប់គ្រងព័ត៌មានសិស្សរួមមាន លេខកូដ ឈ្មោះ ភេទ ថ្ងៃខែឆ្នាំកំណើត លេខទូរស័ព្ទ ថ្នាក់រៀន ហើយនិងផ្នែកគ្រប់គ្រងរួមមានដូចខាងក្រោម

- ផ្នែកបញ្ចូលព័ត៌មាន
- ផ្នែកមើលព័ត៌មាន
- ផ្នែកស្វែងរកព័ត៌មាន
- ផ្នែកលប់ព័ត៌មាន

- ផ្នែកកែសម្រួលព័ត៌មាន

**សំគាល់:** ដោយប្រើ Array ពីរវិមាត្រ ហើយមានការគ្រប់គ្រងក្នុងការបញ្ចូលទិន្នន័យ

```
#include<stdio.h>
```

```
#include<conio.h>
```

```
#include<ctype.h>
```

```
#include<string.h>
```

```
#include<dos.h>
```

```
#define MAX 100
```

```
char stu[MAX][6][30];
```

```
int n,i,j,b;
```

```
char ch;
```

```
char x[20];
```

```
int main_menu( );
```

```
void input( );
```

```
void output( );
```

```
int search_id( char * );
```

```
void delete_id( );
```

```
void edit_id( );
```

```
void inputs( char * );
```

```
void main( ){
```

```
do{
```

```
    clrscr( );
```

```

i=main_menu( );

switch(i){

case 0:

    input( );

    break;

case 1:

    output( );

    break;

case 2:

    textbackground( BLUE );

    clrscr( );

    printf( "Enter student id to Search : " );

    scanf( "%s",&x );

    b=search_id(x);

    if( b==0 )

    {

        printf( "student id not found" );

        getch( );

    }

    break;

case 3:

    delete_id( );

```



```

        break;

    case 4:

        edit_id( );

        break;

    }

}while( i!=5);

}

void input( ){

    textbackground( BLUE );

do{

again:

    clrscr( );

    printf( "Enter student id: " );

    //scanf( "%s",&stu[n][0] );

    inputs( stu[n][0] );

    b=search_id( stu[n][0] );

    if( b!=0 )

    {

        gotoxy( 1,13 );

        printf( "Duplicate id try again: " );

        getch( );

        gotoxy( 1,13 );

```

```

        clrscr( );

        goto again;

    }

    printf( "Enter student name: " );

    scanf( "%s",&stu[n][1] );

    printf( "Enter student sex " );

    scanf( "%s",&stu[n][2] );

    printf( "Enter student dob: " );

    scanf( "%s",&stu[n][3] );

    printf( "Enter student phone: " );

    //scanf( "%s",&stu[n][4] );

    inputs( stu[n][4] );

    printf( "Enter student Class: " );

    scanf( "%s",&stu[n][5] );

    n=n+1;

    printf( "Press Enter continue other exit" );

}while( getch( )==13);

}

void output( ){

textbackground( BLUE );

clrscr( );

if( n<8)

```

```

{

printf( "%10s%15s%10s%15s%15s%10s\n", "ID", "ST_Name", "Sex", "DOB", "Phone", "Class" );

    for( i=0; i<n; i++ )

    {

printf( "%10s%15s%10s%15s%15s%10s\n", stu[i][0],

                                                stu[i][1], stu[i][2], stu[i][3], stu[i][4], stu[i][5] );

    }

    getch( );

}

else

{
    j=0;

    do{

        clrscr( );

printf( "%10s%15s%10s%15s%15s%10s\n", "ID", "ST_Name", "Sex", "DOB", "Phone", "Class" );

        for( i=j; i<j+8; i++ )

        {

printf( "%10s%15s%10s%15s%15s%10s\n", stu[i][0],

                                                stu[i][1], stu[i][2], stu[i][3], stu[i][4], stu[i][5] );

        }

        ch=getch( );

        if( tolower( ch )=='a' )

        {

            j--; if( j<0 ) j++;

```

```

    }

    if( tolower( ch )=='z' )

    {

        j++; if( j+8>n ) j--;

    }

}while( ch !=27 );

}

}

int search_id( char *x){

    b=0;

    for( i=0;i<n;i++)

    {

        if( strcmp( stu[i][0],x)==0 ){ b=1;

        printf( "\nStudent id: %s\n",stu[i][0] );

        printf( "Student name: %s\n",stu[i][1] );

        printf( "Student sex: %s\n",stu[i][2] );

        printf( "Student DoB: %s\n",stu[i][3] );

        printf( "Student Phone:%s\n",stu[i][4] );

        printf( "Student Class:%s\n",stu[i][5] );

        }

    }

    if( b!=0 ) getch( );

```

```

        return b;
    }

void delete_id( ){

    int k;

    textbackground( BLUE );

    clrscr( );

    printf( "Enter student id to delete: " );

    //scanf( "%s",&x );

    inputs( x );

    b=0;

    for( i=0;i<n;i++ )

    {

        if( strcmp( stu[i][0],x)==0 )

        {

            b=1;

            for( j=i;j<n-1;j++ )

            {

                for( k=0;k<6;k++ )

                    strcpy( stu[j][k],stu[j+1][k] );

            }

            n=n-1;

        }

    }

```

```

    }

    if( b==0)

        printf( "Delete not completed" );

    else

        printf( "Delete completed" );

    getch( );

}

void edit_id( ){

    textbackground( BLUE );

    clrscr( );

    printf( "Enter student id to Edit: " );

    //scanf( "%s",&x );

    inputs( x );

    b=0;

    for( i=0;i<n;i++ )

    {

        if( strcmp( stu[i][0],x)==0 )

        {

            b=1;

            printf( "Enter student name: " );

            scanf( "%s",&stu[i][1] );

            printf( "Enter student sex " );

```

```

        scanf( "%s",&stu[i][2] );

        printf( "Enter student dob: " );

        scanf( "%s",&stu[i][3] );

        printf( "Enter student phone: " );

        scanf( "%s",&stu[i][4] );

        printf( "Enter student Class: " );

        scanf( "%s",&stu[i][5] );

    }

}

if( b==0 )

    printf( "Edit not completed" );

else

    printf( "Edit Update completed" );

getch( );

}

int main_menu( ){

do{

    textbackground( BLUE );

    clrscr( );

    textcolor( WHITE );

    textbackground( GREEN );

    gotoxy( 1,10+i);

```

```

cprintf( "%30c", ' ');

gotoxy( 1,10 );

printf( "Register information student" );

gotoxy( 1,11 );

printf( "Show information student" );

gotoxy( 1,12 );

printf( "Search information student" );

gotoxy( 1,13 );

printf( "Delete information student" );

gotoxy( 1,14 );

printf( "Edit information student" );

gotoxy( 1,15 );

printf( "Exit program\n" );

gotoxy( 1,25 );

printf( "Use key to Scroll 'A' move 'UP' or 'Z' move 'Down'" );

printf( " Enter='Select'" );

ch=getch( );

switch( tolower( ch ) ){

case 'a':

    i=i-1;

    if( i<0 ) i=5;

    break;

```



```

        case 'z':

            i=i+1;

            if( i>5 ) i=0;

        }

}while( ch!=27 && ch!=13 );

if( ch==27 ) i=5;

return i;

}

void inputs( char *st){

    do{

        ch=getch( );

        if( ch>='0' && ch<='9' || ch==8 )

        {

            if( ch!=8 )

            {

                st[i]=ch; i++;

                printf( "%c",ch );

            }

            else

            {

                if( i>0 )

                {

```

```

        i--;

        st[i]='\0';

        printf( "%c",ch );

        printf( " " );

        gotoxy( wherex( )-1,wherey( ) );

    }

}

}

}while( ch!=13 );

}

```

## I. និយមន័យ Pointer

Pointer ជាប្រភេទអថេរពិសេសមួយដែលមិនចេះទុកទិន្នន័យខ្លួនឯងទេតែវាចេះប្រើគេអោយទុកទិន្នន័យអោយវា ហើយ Pointer ចេះតែផ្ទុក address ដែលជាប់សំ Location memory ណាមួយដែលវាបាន Point ទៅកាន់។ ខាងក្រោមជាការបង្ហាញពី Pointer

Syntax:

Datatype \*identifier;

Ex:

```
int    *ip;        /* pointer to an integer */
double *dp;        /* pointer to a double */
float  *fp;        /* pointer to a float */
char   *ch         /* pointer to a character */
```

## II. របៀបប្រើ Pointer

Pointer មិនចេះទុកទិន្នន័យទេដើម្បីអោយ Pointer ប្រើគេផ្ទុកទិន្នន័យអោយវាបានយើងត្រូវដឹងរបៀបប្រើ Pointer ចង្អុលទៅកាន់អ្នកដែលអាចអោយវាផ្ទុកទិន្នន័យបានជាមុនសិន។ ខាងក្រោមជារបៀបប្រើ Pointer ទៅកាន់អ្នកដែលអាចអោយ Pointer ផ្ទុកទិន្នន័យបាន។

Ex:

```
int *p;
int a;
p = &a;
```

## ខាងក្រោមជាឧទាហរណ៍នៃការប្រើ Pointer

```
#include <stdio.h>
```

```
int main ( ) {
```

```
    int var = 20; /* actual variable declaration */
```

```
    int *ip;      /* pointer variable declaration */
```

```

ip = &var; /* store address of var in pointer variable */

printf( "Address of var variable: %x\n", &var );

/* address stored in pointer variable */

printf( "Address stored in ip variable: %x\n", ip );

/* access the value using the pointer */

printf( "Value of *ip variable: %d\n", *ip );

return 0;

}

```

### ខាងក្រោមជាលទ្ធផលកម្មវិធីខាងលើ

Address of var variable: bffd8b3c

Address stored in ip variable: bffd8b3c

Value of \*ip variable: 20

### III. របៀបប្រើ Pointer arithmetic

Pointer មិនមែនជាអ្នកផ្ទុកតម្លៃទេ Pointer ចេះតែផ្ទុក Address ដូច្នេះការធ្វើប្រមាណវិធីជាមួយ Pointer គឺអាចប្រើបានតែសញ្ញាប្រមាណវិធីបូកឬដក (++, --) ដែលមានន័យថាជាការរំកិល Address របស់ Location Memory ណាមួយដែលវាកំពុងតែ Point ទៅកាន់។ ខាងក្រោមជាការបង្ហាញអំពីរបៀប

```

#include <stdio.h>

const int MAX = 3;

int main ( ) {
    int var[] = {10, 100, 200};
    int i, *ptr;

    /* let us have array address in pointer */
    ptr = var;

    for ( i = 0; i < MAX; i++ ) {

```

```

printf( "Address of var[%d] = %x\n", i, ptr );
printf( "Value of var[%d] = %d\n", i, *ptr );
/* move to the next location */
ptr++;
}
return 0;
}

```

### **ខាងក្រោមជាលទ្ធផលនៃកម្មវិធីខាងលើ**

Address of var[0] = bf882b30

Value of var[0] = 10

Address of var[1] = bf882b34

Value of var[1] = 100

Address of var[2] = bf882b38

Value of var[2] = 200

### **ខាងក្រោមជាវិធីសាស្ត្រផ្សេងមួយទៀតក្នុងការប្រើ**

```
#include <stdio.h>
```

```
const int MAX = 3;
```

```
int main ( ) {
```

```
    int var[] = {10, 100, 200};
```

```
    int i, *ptr;
```

```
    /* let us have array address in pointer */
```

```
    ptr = &var[MAX-1];
```

```
    for ( i = MAX; i > 0; i-- )
```

```
    {
```

```
        printf( "Address of var[%d] = %x\n", i-1, ptr );
```

```
        printf( "Value of var[%d] = %d\n", i-1, *ptr );
```

```

/* move to the previous location */

ptr--;

}

return 0;

}

```

### **ខាងក្រោមជាលទ្ធផលនៃកម្មវិធីខាងលើ**

Address of var[2] = bfedbcd8

Value of var[2] = 200

Address of var[1] = bfedbcd4

Value of var[1] = 100

Address of var[0] = bfedbcd0

Value of var[0] = 10

## **IV. របៀបប្រើ Pointer និង Array**

Pointer និង Array មុននឹងយល់អំពីការប្រើ ។ ចូរមើលការបង្ហាញនៃការប្រើ Array ជាចំនួនគត់ដែរមានបីធាតុដូចខាងក្រោម

```

#include <stdio.h>

const int MAX = 3;

int main ( ) {
    int var[] = {10, 100, 200};
    int i;

    for ( i = 0; i < MAX; i++ ) {
        printf( "Value of var[%d] = %d\n", i, var[i] );
    }

    return 0;
}

```

```
}
```

### **ខាងក្រោមជាលទ្ធផលកម្មវិធីខាងលើ**

Value of var[0] = 10

Value of var[1] = 100

Value of var[2] = 200

វាអាចមានស្ថានភាពច្រើននៅពេលដែលយើងចង់ប្រើ Pointer ទៅ Array ជាចំនួនគត់ ឬ char ឬប្រភេទទិន្នន័យផ្សេងទៀតដែលមាន។ ខាងក្រោមនេះគឺជាការប្រកាស Pointer ទៅ Array ចំនួនគត់ –

Syntax:

```
int *p[size];
```

### **ខាងក្រោមជាឧទាហរណ៍នៃការប្រើ Pointer ទៅ Array ចំនួនគត់**

```
#include <stdio.h>
```

```
const int MAX = 3;
```

```
int main ( ) {
```

```
    int var[] = {10, 100, 200};
```

```
    int i, *ptr[MAX];
```

```
    for ( i = 0; i < MAX; i++ )
```

```
{
```

```
        ptr[i] = &var[i]; /* assign the address of integer. */
```

```
}
```

```
    for ( i = 0; i < MAX; i++ ) {
```

```
        printf( "Value of var[%d] = %d\n", i, *ptr[i] );
```

```
}
```

```
    return 0;
```

```
}
```

## ខាងក្រោមជាលទ្ធផលកម្មវិធីខាងលើ

Value of var[0] = 10

Value of var[1] = 100

Value of var[2] = 200

## ខាងក្រោមជាឧទាហរណ៍នៃការប្រើ Pointer ទៅ Array ក្នុងកម្មវិធី

```
#include <stdio.h>
```

```
const int MAX = 4;
```

```
int main ( ) {
```

```
    char *names[] = {
```

```
        "Zara Ali",
```

```
        "Hina Ali",
```

```
        "Nuha Ali",
```

```
        "Sara Ali"
```

```
    };
```

```
    int i = 0;
```

```
    for ( i = 0; i < MAX; i++ ) {
```

```
        printf( "Value of names[%d] = %s\n", i, names[i] );
```

```
    }
```

```
    return 0;
```

```
}
```

## ខាងក្រោមជាលទ្ធផលកម្មវិធីខាងលើ

Value of names[0] = Zara Ali



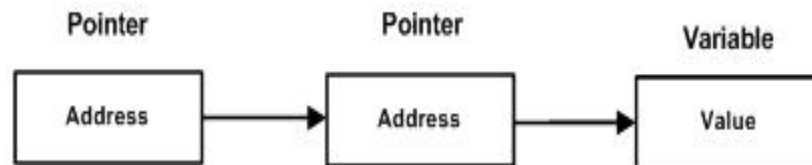
Value of names[1] = Hina Ali

Value of names[2] = Nuha Ali

Value of names[3] = Sara Ali

## V. របៀបប្រើ Pointer និង Pointer

Pointer និង Pointer ជាការប្រើអោយ Pointer ចង្អុលទៅកាន់ Pointer គ្នាវាបន្តទៀត។ ខាងក្រោមជាការបង្ហាញនៃការប្រើ



Ex:

```
int **var;
```

នៅពេលដែលអ្នកទុកតម្លៃគោលដៅប្រើត្រូវបានចង្អុលបង្ហាញដោយ Pointer ណាមួយហើយការចូលប្រើតម្លៃនោះតម្រូវអោយមានការប្រើសញ្ញាផ្តោតពីរដង ដូចដែលបានបង្ហាញខាងក្រោមក្នុងឧទាហរណ៍

```
#include <stdio.h>
```

```
int main ( ) {
```

```
    int var;
```

```
    int *ptr;
```

```
    int **pptr;
```

```
    var = 3000;
```

```
    /* take the address of var */
```

```
    ptr = &var;
```

```
    /* take the address of ptr using address of operator & */
```

```
    pptr = &ptr;
```

```
    /* take the value using pptr */
```

```

printf( "Value of var = %d\n", var );

printf( "Value available at *ptr = %d\n", *ptr );

printf( "Value available at **pptr = %d\n", **pptr );

return 0;

}

```

## ខាងក្រោមជាលទ្ធកម្មវិធីខាងលើ

Value of var = 3000

Value available at \*ptr = 3000

Value available at \*\*pptr = 3000

## VI. របៀបប្រើ Pointer និង Function

### 1. Pointer ជា Parameter របស់ Function

នៅពេលដែល Function មាន Parameter ហើយជា Pointer យើងមានរបៀបប្រើដូចការបង្ហាញខាងក្រោម

```

#include <stdio.h>

#include <time.h>

void getSeconds( unsigned long *par );

int main ( ) {
    unsigned long sec;
    getSeconds( &sec );
    /* print the actual value */
    printf( "Number of seconds: %ld\n", sec );
    return 0;
}

void getSeconds( unsigned long *par ) {
    /* get the current number of seconds */
    *par = time( NULL );
    return;
}

```

## ខាងក្រោមជាលទ្ធកម្មវិធីខាងលើ

Number of seconds :1294450468

## ការបង្ហាញម្យ៉ាងទៀតពី Function មាន Parameter ជា Pointer ក៏ដូចជា Array

```
#include <stdio.h>

/* function declaration */
double getAverage( int *arr, int size );

int main ( ) {
    /* an int array with 5 elements */
    int balance[5] = {1000, 2, 3, 17, 50};
    double avg;

    /* pass pointer to the array as an argument */
    avg = getAverage( balance, 5 );

    /* output the returned value */
    printf( "Average value is: %f\n", avg );

    return 0;
}

double getAverage( int *arr, int size ) {
    int i, sum = 0;
    double avg;

    for ( i = 0; i < size; ++i )
    {
        sum += arr[i];
    }

    avg = ( double )sum / size;

    return avg;
}
```

## ខាងក្រោមជាលទ្ធកម្មវិធីខាងលើ

Average value is: 214.40000

## 2. Function មាន return ជា Pointer

ការពីមេរៀន Function យើងបានឃើញនៃការប្រើ Function return Pointer ហើយដូចទំរង់ខាងក្រោម

Syntax:

```
int * myFunction( ) {  
    .  
    .  
    .  
}
```

ចំណុចសំខាន់មួយទៀតដែលត្រូវចងចាំគឺថា វាមិនមែនជាគំនិតល្អទេក្នុងការប្រើ Function return តាមរយៈ Pointer អ្នកត្រូវការអថេរមូលដ្ឋាននៅខាងក្រៅ Function ចាំទទួល។ដូច្នេះអ្នកនឹងត្រូវកំណត់អថេរមូលដ្ឋានជាអថេរ Static ទើបប្រសើរ។ខាងក្រោមជាឧទាហរណ៍បញ្ជាក់

```
#include <stdio.h>
```

```
#include <time.h>
```

```
/* function to generate and return random numbers. */
```

```
int * getRandom( ) {
```

```
    static int r[10];
```

```
    int i;
```

```
    /* set the seed */
```

```
    srand( (unsigned)time( NULL ) );
```

```
    for ( i = 0; i < 10; ++i) {
```

```
        r[i] = rand( );
```

```
        printf( "%d\n", r[i] );
```

```
    }
```

```
    return r;
```

```
}
```

```

/* main function to call above defined function */

int main ( ) {

    /* a pointer to an int */

    int *p;

    int i;

    p = getRandom( );

    for ( i = 0; i < 10; i++ ) {

        printf( "(p + [%d]) : %d\n", i, *(p + i) );

    }

    return 0;

}

```

### **ខាងក្រោមជាលទ្ធផលរបស់កម្មវិធីខាងលើ**

1523198053

1187214107

1108300978

430494959

1421301276

930971084

123250484

106932140

1604461820

149169022

$*(p + [0]) : 1523198053$

$*(p + [1]) : 1187214107$

$*(p + [2]) : 1108300978$

$*(p + [3]) : 430494959$

$*(p + [4]) : 1421301276$

$*(p + [5]) : 930971084$

$*(p + [6]) : 123250484$

$*(p + [7]) : 106932140$

$*(p + [8]) : 1604461820$

$*(p + [9]) : 149169022$

## VII. របៀបប្រើ Pointer និង Memory Management

ចំនុចនេះពន្យល់អំពីការគ្រប់គ្រង dynamic memory នៅក្នុងភាសា C ។ ភាសាសរសេរកម្មវិធី C ផ្តល់នូវមុខងារជាច្រើនសម្រាប់ការបែងចែក និងការគ្រប់គ្រង memory។ មុខងារទាំងនេះអាចរកបាននៅក្នុង Library `<stdlib.h>` ។ ខាងក្រោមជាឃ្លាបញ្ជាដែលប្រើក្នុងការគ្រប់គ្រង Memory

| Sr.No. | Function & ការពិពណ៌នា                                                                                    |
|--------|----------------------------------------------------------------------------------------------------------|
| 1      | <code>void *calloc(int num, int size);</code><br>ឃ្លាបញ្ជានេះប្រើសំរាប់ដំណើរការ Location memory ជា Array |
| 2      | <code>void free(void *address);</code><br>ឃ្លាបញ្ជានេះប្រើសំរាប់សំអាត Location memory                    |
| 3      | <code>void *malloc(size_t size);</code>                                                                  |

|   |                                                                                                                   |
|---|-------------------------------------------------------------------------------------------------------------------|
|   | ឃ្លាបញ្ជានេះប្រើសំរាប់ដំណើរការ Location memory ជាតម្លៃណាមួយ                                                       |
| 4 | <b>void *realloc( void *address, int newsize );</b><br>ឃ្លាបញ្ជានេះប្រើសំរាប់ដំណើរការបន្ថែមឬបន្ថយ Location memory |

## 1. ការប្រើ malloc

### ខាងក្រោមជាឧទាហរណ៍នៃការប្រើឃ្លាបញ្ជា malloc

```
#include <stdio.h>
#include <stdlib.h>
#include <string.h>

int main( ) {
    char name[100];
    char *description;
    strcpy( name, "Zara Ali" );
    /* allocate memory dynamically */
    description = malloc( 200 * sizeof( char ) );
    if( description == NULL ) {
        fprintf( stderr, "Error - unable to allocate required memory\n" );
    } else {
        strcpy( description, "Zara ali a DPS student in class 10th" );
    }
    printf( "Name = %s\n", name );
    printf( "Description: %s\n", description );
}
```

### ខាងក្រោមជាលទ្ធផលនៃកម្មវិធីខាងលើ

Name = Zara Ali  
 Description: Zara ali a DPS student in class 10th

## 2. ការប្រើ calloc

### ខាងក្រោមជាឧទាហរណ៍នៃការប្រើឃ្លាបញ្ជា calloc

```
#include <stdio.h>
```

```

#include <stdlib.h>
#include <string.h>

int main( ) {
    char name[100];
    char *description;
    strcpy( name, "Zara Ali" );
    /* allocate memory dynamically */
    description = calloc( 200 , sizeof( char ) );
    if( description == NULL ) {
        fprintf( stderr, "Error - unable to allocate required memory\n" );
    } else {
        strcpy( description, "Zara ali a DPS student in class 10th" );
    }
    printf( "Name = %s\n", name );
    printf( "Description: %s\n", description );
}

```

**ខាងក្រោមជាលទ្ធផលនៃកម្មវិធីខាងលើ**

Name = Zara Ali

Description: Zara ali a DPS student in class 10th

### 3. ការប្រើ realloc

ខាងក្រោមជាឧទាហរណ៍នៃការប្រើប្រាស់ realloc

```

#include <stdio.h>
#include <stdlib.h>
#include <string.h>

int main( ) {
    char name[100];
    char *description;
    strcpy( name, "Zara Ali" );
    /* allocate memory dynamically */
    description = malloc( 30 * sizeof( char ) );

```



```

if( description == NULL ) {
    fprintf( stderr, "Error - unable to allocate required memory\n" );
} else {
    strcpy( description, "Zara ali a DPS student." );
}
/* suppose you want to store bigger description */
description = realloc( description, 100 * sizeof( char ) );
if( description == NULL ) {
    fprintf( stderr, "Error - unable to allocate required memory\n" );
} else {
    strcat( description, "She is in class 10th" );
}
printf( "Name = %s\n", name );
printf( "Description: %s\n", description );
/* release memory using free( ) function */
free( description );
}

```

### **ខាងក្រោមជាលទ្ធផលនៃកម្មវិធីខាងលើ**

Name = Zara Ali

Description: Zara ali a DPS student.She is in class 10th

## មេរៀនទី១១

## ការប្រើប្រាស់ Structure

### I. និយមន័យ Structure

Structure ជាប្រភេទអថេរពិសេសមួយអាចប្រមូលផ្តុំប្រភេទទិន្នន័យជាច្រើនដាក់បញ្ចូលគ្នាជាប្រភេទទិន្នន័យតែមួយ។ ដោយធាតុរបស់ Structure គេហើយថា File ដែលអាចមានឈ្មោះនិងប្រភេទទិន្នន័យខុសៗគ្នា។ ខាងក្រោមជាទំរង់ទូទៅនៃការបង្កើត Structure

Syntax:

```
struct [structure tag] {  
    datatype Field;  
    datatype Field;  
    ...  
    datatype Field;  
} [one or more structure variables];
```

- [structure tag] : ជាឈ្មោះរបស់ structure type ដែលយើងនឹងបង្កើត
- Field : ជាឈ្មោះនៃធាតុរបស់ Structure
- Structure variables: ជាអថេរ Structure

### II. ការបង្កើត Structure

ការបង្កើត Structure ជាប្រភេទទិន្នន័យអ្វីមួយមានន័យថាទិន្នន័យនោះជាសំនុំដែលមានធាតុដែលមានប្រភេទផ្សេងៗគ្នាខាងក្រោមជាឧទាហរណ៍បង្ហាញអំពីការបង្កើត Structure Datatype

Ex:

```
struct Books {  
    char title[50];  
    char author[50];  
    char subject[100];  
    int book_id;  
};
```

Books book;

**ខាងក្រោមជាឧទាហរណ៍បង្ហាញអំពីការបង្កើត Structure variable**

Ex:

```
struct {
```

```

char title[50];
char author[50];
char subject[100];
int book_id;
} book;

```

### III. របៀបប្រើ Structure

របៀបប្រើ Structure ក្រោយពីប្រកាសអថេរ Structure រួចត្រូវការផ្តល់តម្លៃអោយទៅកាន់ធាតុរបស់ Structure ណាមួយទុកនោះត្រូវសរសេរឈ្មោះអថេរ Structure ជាមួយនិងសញ្ញា "." ចុចបន្ទាប់មកជាឈ្មោះ Field របស់ Structure ដែរនិងទទួលយកទិន្នន័យ។ ខាងក្រោមជាឧទាហរណ៍នៃការប្រើ

```

#include <stdio.h>
#include <string.h>
struct Books {
    char title[50];
    char author[50];
    char subject[100];
    int book_id;
};
int main( ) {
    struct Books Book1;    /* Declare Book1 of type Book */
    struct Books Book2;    /* Declare Book2 of type Book */
    /* book 1 specification */
    strcpy( Book1.title, "C Programming" );
    strcpy( Book1.author, "Nuha Ali" );
    strcpy( Book1.subject, "C Programming Tutorial" );
    Book1.book_id = 6495407;
    /* book 2 specification */
    strcpy( Book2.title, "Telecom Billing" );
    strcpy( Book2.author, "Zara Ali" );
    strcpy( Book2.subject, "Telecom Billing Tutorial" );
    Book2.book_id = 6495700;
}

```

```

/* print Book1 info */
printf( "Book 1 title : %s\n", Book1.title );
printf( "Book 1 author : %s\n", Book1.author );
printf( "Book 1 subject : %s\n", Book1.subject );
printf( "Book 1 book_id : %d\n", Book1.book_id );
/* print Book2 info */
printf( "Book 2 title : %s\n", Book2.title );
printf( "Book 2 author : %s\n", Book2.author );
printf( "Book 2 subject : %s\n", Book2.subject );
printf( "Book 2 book_id : %d\n", Book2.book_id );
return 0;
}

```

### **ខាងក្រោមជារង្វង់ផលនៃកម្មវិធីខាងលើ**

Book 1 title : C Programming

Book 1 author : Nuha Ali

Book 1 subject : C Programming Tutorial

Book 1 book\_id : 6495407

Book 2 title : Telecom Billing

Book 2 author : Zara Ali

Book 2 subject : Telecom Billing Tutorial

Book 2 book\_id : 6495700

#### **IV. របៀបប្រើ Structure និង Array**

របៀបប្រើ Structure និង Array នៅពេលដែលត្រូវការផ្ទុកទិន្នន័យច្រើននោះយើងត្រូវចេះប្រើវានិង Array ។ ខាងក្រោមបង្ហាញនៃការប្រើ

```
#include <stdio.h>
```

```
#include <string.h>
```

```
#define MAX 100
```

```
struct Books {
```

```
    char title[50];
```

```
    char author[50];
```

```

char subject[100];
int book_id;
};
int main( ) {
    Books book[MAX];
    /* book 1 specification */
    strcpy( book[0].title, "C Programming");
    strcpy( book[0].author, "Nuha Ali");
    strcpy( book[0].subject, "C Programming Tutorial");
    book[0].book_id = 6495407;
    /* book 2 specification */
    strcpy( book[1].title, "Telecom Billing");
    strcpy( book[1].author, "Zara Ali");
    strcpy( book[1].subject, "Telecom Billing Tutorial");
    Book2.book_id = 6495700;
    /* print Book1 info */
    printf( "Book 1 title : %s\n", book[0].title);
    printf( "Book 1 author : %s\n", book[0].author);
    printf( "Book 1 subject : %s\n", book[0].subject);
    printf( "Book 1 book_id : %d\n", book[0].book_id);
    /* print Book2 info */
    printf( "Book 2 title : %s\n", book[1].title);
    printf( "Book 2 author : %s\n", book[1].author);
    printf( "Book 2 subject : %s\n", book[1].subject);
    printf( "Book 2 book_id : %d\n", book[1].book_id);
    return 0;
}

```

### **ខាងក្រោមជានឡូផលនៃកម្មវិធីខាងលើ**

Book 1 title : C Programming

Book 1 author : Nuha Ali

Book 1 subject : C Programming Tutorial

Book 1 book\_id : 6495407

Book 2 title : Telecom Billing

Book 2 author : Zara Ali

Book 2 subject : Telecom Billing Tutorial

Book 2 book\_id : 6495700

## V. របៀបប្រើ Structure និង Pointer

Structure អាចប្រើជាប្រភេទ Pointer ដែរមានទំរង់នៃការប្រើដូចខាងក្រោម

Syntax:

```
Structure_type *Var_name;
```

```
Var_name->Field = value;
```

Or

```
( *Var_name ).Field = value;
```

ខាងក្រោមជាឧទាហរណ៍បង្ហាញនៃការប្រើ

```
#include <stdio.h>
```

```
#include <string.h>
```

```
struct Books {
```

```
    char title[50];
```

```
    char author[50];
```

```
    char subject[100];
```

```
    int book_id;
```

```
};
```

```
int main( ) {
```

```
    struct Books *Book1;    /* Declare Book1 of type Book */
```

```
    struct Books *Book2;    /* Declare Book2 of type Book */
```

```
    /* book 1 specification */
```

```
    strcpy( Book1->title, "C Programming");
```

```
    strcpy( Book1->author, "Nuha Ali");
```

```
    strcpy( Book1->subject, "C Programming Tutorial");
```

```

Book1->book_id = 6495407;
/* book 2 specification */
strcpy( Book2->title, "Telecom Billing" );
strcpy( Book2->author, "Zara Ali" );
strcpy( Book2->subject, "Telecom Billing Tutorial" );
Book2->book_id = 6495700;
/* print Book1 info */
printf( "Book 1 title : %s\n", Book1->title );
printf( "Book 1 author : %s\n", Book1->author );
printf( "Book 1 subject : %s\n", Book1->subject );
printf( "Book 1 book_id : %d\n", Book1->book_id );
/* print Book2 info */
printf( "Book 2 title : %s\n", Book2->title );
printf( "Book 2 author : %s\n", Book2->author );
printf( "Book 2 subject : %s\n", Book2->subject );
printf( "Book 2 book_id : %d\n", Book2->book_id );
return 0;
}

```

### **ខាងក្រោមជារង្វង់ផលនៃកម្មវិធីខាងលើ**

Book 1 title : C Programming  
 Book 1 author : Nuha Ali  
 Book 1 subject : C Programming Tutorial  
 Book 1 book\_id : 6495407  
 Book 2 title : Telecom Billing  
 Book 2 author : Zara Ali  
 Book 2 subject : Telecom Billing Tutorial  
 Book 2 book\_id : 6495700

## **VI. របៀបប្រើ Structure និង Function**

របៀបប្រើ Structure ជា Parameter របស់ Function មានរបៀបប្រើដូចឧទាហរណ៍ខាងក្រោម

```

#include <stdio.h>
#include <string.h>
struct Books {
    char title[50];
    char author[50];
    char subject[100];
    int book_id;
};
/* function declaration */
void printBook( struct Books *book );
int main( ) {
    struct Books Book1;    /* Declare Book1 of type Book */
    struct Books Book2;    /* Declare Book2 of type Book */

    /* book 1 specification */
    strcpy( Book1.title, "C Programming" );
    strcpy( Book1.author, "Nuha Ali" );
    strcpy( Book1.subject, "C Programming Tutorial" );
    Book1.book_id = 6495407;
    /* book 2 specification */
    strcpy( Book2.title, "Telecom Billing" );
    strcpy( Book2.author, "Zara Ali" );
    strcpy( Book2.subject, "Telecom Billing Tutorial" );
    Book2.book_id = 6495700;
    /* print Book1 info by passing address of Book1 */
    printBook( &Book1 );
    /* print Book2 info by passing address of Book2 */
    printBook( &Book2 );
    return 0;
}
void printBook( struct Books *book ) {

```



```

printf( "Book title : %s\n", book->title );
printf( "Book author : %s\n", book->author );
printf( "Book subject : %s\n", book->subject );
printf( "Book book_id : %d\n", book->book_id );
}

```

### **ខាងក្រោមជាលទ្ធផលរបស់កម្មវិធីខាងលើ**

```

Book title : C Programming
Book author : Nuha Ali
Book subject : C Programming Tutorial
Book book_id : 6495407
Book title : Telecom Billing
Book author : Zara Ali
Book subject : Telecom Billing Tutorial
Book book_id : 6495700

```

## **VII. ការប្រើ Search**

នៅពេលមានទិន្នន័យច្រើនដែលផ្ទុកនៅក្នុង Structure យើងអាចប្រើ Array Structure នោះវិធីសាស្ត្រ Search និងត្រូវប្រើនៅពេលដែលត្រូវការស្វែងរកទិន្នន័យ។ខាងក្រោមជាឧទាហរណ៍បង្ហាញនៃការប្រើ

```

#include <stdio.h>
#include<conio.h>
#include <string.h>
#define MAX 100
struct Books {
    char title[50];
    char author[50];
    char subject[100];
    char book_id[20];
};
Books book[MAX];
void main( ){    int i,b,n;
                char x[20];

```

```

clrscr( );
printf( "Enter number book : " );
scanf( "%d",&n);
for( i=0 ; i<n; i++){
printf( "Enter id book: " );
gets( book[i].book_id );
printf( "Enter title book: " );
gets( book[i].title );
printf( "Enter author book: " );
gets( book[i].author );
printf( "Enter subject book: " );
gets( book[i].subject );
}
printf( "Enter id book to search: " );
gets( x );
b=0;
for( i=0; i<n; i++ )
{
    if( strcmp( book[i].book_id,x)==0 )
    {
        b=1;

        printf( "id book : %s\n",book[i].book_id );
        printf( "title book : %s\n",book[i].title );
        printf( "author book: %s\n",book[i].author );
        printf( "subject book: %s\n",book[i].subject );
    }
}

```

```

        if( b==0 )

            printf( "book id not found" );

        getch( );

    }

```

### **ខាងក្រោមជាលទ្ធផល១របស់កម្មវិធីខាងលើ**

```

Enter number book : 2
Enter id book: b001
Enter title book: c program
Enter author book: Mr.C
Enter subject book: programming
Enter id book: b002
Enter title book: Math program
Enter author book: Mr.Math
Enter subject book: Math
Enter id book to search: b000
book id not found

```

### **ខាងក្រោមជាលទ្ធផល២របស់កម្មវិធីខាងលើ**

```

Enter number book : 2
Enter id book: b001
Enter title book: c program
Enter author book: Mr.C
Enter subject book: programming
Enter id book: b002
Enter title book: Math program
Enter author book: Mr.Math
Enter subject book: Math
Enter id book to search: b001
id book : b001
title book: c program

```

author book: Mr.C

subject book: programming

### VIII. ការប្រើ Sort

នៅពេលមានទិន្នន័យច្រើនដែលផ្ទុកនៅក្នុង Structure យើងអាចប្រើ Array Structure នោះវិធីសាស្ត្រ Sort និង ត្រូវប្រើនៅពេលដែលត្រូវការតម្រៀបទិន្នន័យអោយមានលំដាប់។ខាងក្រោមជាឧទាហរណ៍បង្ហាញនៃការប្រើ

```
#include <stdio.h>
#include<conio.h>
#include <string.h>
struct Books {
    char  book_id[20];
    char  title[50];
    char  author[50];
    char  subject[100];
};
Books book[3]={
    {"b001","math lab","Mr.A","math"},
    {"b002","networking","Mr.B","network"},
    {"b003","c program","Mr.C","program"}
};

Books BK;

void main( ){    int i,j,b,n;
                char x[20];

                clrscr( );
                printf( "before sort information\n" );
                printf( "%10s%20s%10s%10s\n","Book_id","Book_title","Author","Subject" );
                for( i=0; i<3; i++ )
                {
                    printf( "%10s%20s%10s%10s\n",book[i].book_id,
```

```

        book[i].title, book[i].author,book[i].subject);

    }

    for( i=0; i<n; i++)

    {

        for( j=i+1; j<n; j++)

        {

            if( strcmp( book[i].title,book[j].title )>0 )

            {

                BK = book[i];

                book[i] = book[j];

                book[j] = BK;

            }

        }

    }

    printf( "after sort information\n" );

    printf( "%10s%20s%10s%10s\n","Book_id","Book_title","Author","Subject" );

    for( i=0; i<3; i++)

    {

        printf( "%10s%20s%10s%10s\n",book[i].book_id,

                book[i].title, book[i].author,book[i].subject);

    }

    getch( );

}

```

**ខាងក្រោមជាលទ្ធផលរបស់កម្មវិធីខាងលើ**

before sort information

| Book_id | Book_title | Author | Subject |
|---------|------------|--------|---------|
| b001    | math lab   | Mr.A   | math    |
| b002    | networking | Mr.B   | network |
| b003    | c program  | Mr.C   | program |

after sort information

| Book_id | Book_title | Author | Subject |
|---------|------------|--------|---------|
| b003    | c program  | Mr.C   | program |
| b001    | math lab   | Mr.A   | math    |
| b002    | networking | Mr.B   | network |

## IX. ការប្រើ Delete

នៅពេលមានទិន្នន័យច្រើនដែលផ្ទុកនៅក្នុង Structure យើងអាចប្រើ Array Structure នោះវិធីសាស្ត្រ Delete និងត្រូវប្រើនៅពេលដែលត្រូវការលេបទិន្នន័យណាមួយ។ខាងក្រោមជាឧទាហរណ៍បង្ហាញនៃការប្រើ

```
#include <stdio.h>
#include<conio.h>
#include <string.h>
struct Books {
    char book_id[20];
    char title[50];
    char author[50];
    char subject[100];
};
Books book[3]={
    {"b001","math lab","Mr.A","math"},
    {"b002","networking","Mr.B","network"},
    {"b003","c program","Mr.C","program"}
};

Books BK;
```

```

void main( ){    int i,j,b,n;
                char x[20];
                clrscr( );
                printf( "before delete information\n" );
                printf( "%10s%20s%10s%10s\n", "Book_id", "Book_title", "Author", "Subject" );
                for( i=0; i<3; i++ )
                {
                printf( "%10s%20s%10s%10s\n", book[i].book_id,
                                book[i].title, book[i].author, book[i].subject );

                }

                printf( "please enter book id to delete: " );

                gets( x );

                b=0;

                for( i=0; i<n; i++ )

                {

                        if( strcmp( book[i].book_id, x ) == 0 )

                        {

                                b=1;

                                for( j=i; j<n; j++ )

                                {

                                        book[j] = book[j+1];

                                }

                                n--;

                        }

                }

```

```

    }

    if( b==0 )

        printf( "Delete not completed" );

    else

        printf( "Delete completed" );

    printf( "after delete information\n" );
    printf( "%10s%20s%10s%10s\n", "Book_id", "Book_title", "Author", "Subject" );
    for( i=0; i<3; i++ )
    {
        printf( "%10s%20s%10s%10s\n", book[i].book_id,

                                                    book[i].title, book[i].author, book[i].subject );

    }

    getch( );
}

```

### **ខាងក្រោមជាលទ្ធផលរបស់កម្មវិធីខាងលើ**

before delete information

| Book_id | Book_title | Author | Subject |
|---------|------------|--------|---------|
| b001    | math lab   | Mr.A   | math    |
| b002    | networking | Mr.B   | network |
| b003    | c program  | Mr.C   | program |

please enter book id to delete: b001

Delete completed

after delete information

| Book_id | Book_title | Author | Subject |
|---------|------------|--------|---------|
| b002    | networking | Mr.B   | network |
| b003    | c program  | Mr.C   | program |



## ខាងក្រោមជាលទ្ធផល២របស់កម្មវិធីខាងលើ

before delete information

| Book_id | Book_title | Author | Subject |
|---------|------------|--------|---------|
| b001    | math lab   | Mr.A   | math    |
| b002    | networking | Mr.B   | network |
| b003    | c program  | Mr.C   | program |

please enter book id to delete: b000

Delete not completed

after delete information

| Book_id | Book_title | Author | Subject |
|---------|------------|--------|---------|
| b001    | math lab   | Mr.A   | math    |
| b002    | networking | Mr.B   | network |
| b003    | c program  | Mr.C   | program |

## X. ការប្រើ Edit

នៅពេលមានទិន្នន័យច្រើនដែលផ្ទុកនៅក្នុង Structure យើងអាចប្រើ Array Structure នោះវិធីសាស្ត្រEdit និង ត្រូវប្រើនៅពេលដែលត្រូវការកែសម្រួលទិន្នន័យណាមួយ។ខាងក្រោមជាឧទាហរណ៍បង្ហាញនៃការប្រើ

```
#include <stdio.h>
```

```
#include<conio.h>
```

```
#include <string.h>
```

```
struct Books {
```

```
    char  book_id[20];
```

```
    char title[50];
```

```
    char author[50];
```

```
    char subject[100];
```

```
};
```

```
Books book[3]={
```

```
    {"b001","math lab","Mr.A","math"},
```

```
    {"b002","networking","Mr.B","network"},
```

```
    {"b003","c program","Mr.C","program"}
```

```

};

Books BK;

void main( ){   int i,j,b,n;

                char x[20];

                clrscr( );

                printf( "before edit information\n" );

                printf( "%10s%20s%10s%10s\n", "Book_id", "Book_title", "Author", "Subject" );

                for( i=0; i<3; i++ )

                {

                printf( "%10s%20s%10s%10s\n", book[i].book_id,

                book[i].title, book[i].author, book[i].subject );

                }

                printf( "please enter book id to edit " );

                gets( x );

                b=0;

                for( i=0; i<n; i++ )

                {

                        if( strcmp( book[i].book_id, x ) == 0 )

                        {

                                b=1;

                                printf( "Enter new id book: " );

                                gets( book[i].book_id );

                                printf( "Enter new title book: " );

                                gets( book[i].title );

                                printf( "Enter new author book: " );

                                gets( book[i].author );

                                printf( "Enter new subject book: " );

                                gets( book[i].subject );

                        }

                }

                if( b == 0 )

```

```

        printf( "Edit not completed" );
else
    printf( "Edit update completed" );
printf( "after edit information\n" );
    printf( "%10s%20s%10s%10s\n", "Book_id", "Book_title", "Author", "Subject" );
    for( i=0; i<3; i++ )
    {
        printf( "%10s%20s%10s%10s\n", book[i].book_id,
            book[i].title, book[i].author, book[i].subject );
    }
    getch( );
}

```

### ខាងក្រោមជាលទ្ធផលរបស់កម្មវិធីខាងលើ

before edit information

| Book_id | Book_title | Author | Subject |
|---------|------------|--------|---------|
| b001    | math lab   | Mr.A   | math    |
| b002    | networking | Mr.B   | network |
| b003    | c program  | Mr.C   | program |

please enter book id to edit: b001

Enter new id book: b001

Enter new title book: math\_lab

Enter new author book: Mr.Math

Enter new subject book: math

after delete information

| Book_id | Book_title | Author  | Subject |
|---------|------------|---------|---------|
| b001    | math_lab   | Mr.Math | math    |
| b002    | networking | Mr.B    | network |
| b003    | c program  | Mr.C    | program |

### ខាងក្រោមជាលទ្ធផល២របស់កម្មវិធីខាងលើ

before edit information

| Book_id | Book_title | Author | Subject |
|---------|------------|--------|---------|
| b001    | math lab   | Mr.A   | math    |
| b002    | networking | Mr.B   | network |
| b003    | c program  | Mr.C   | program |

please enter book id to edit: b000

edit not completed

after edit information

| Book_id | Book_title | Author | Subject |
|---------|------------|--------|---------|
| b001    | math lab   | Mr.A   | math    |
| b002    | networking | Mr.B   | network |
| b003    | c program  | Mr.C   | program |

#### XI. លំហាត់ការគ្រប់គ្រងផ្នែកលក់ទំនិញក្នុងហាងមួយ

កម្មវិធីហាងលក់ទំនិញមានផ្នែកច្រើនក្នុងការគ្រប់គ្រងកម្មវិធីខាងក្រោមលើកយកតែផ្នែកលក់មកបង្ហាញរួមមានការងារដូចជា

- ផ្នែក Stock product
  - o Input product
  - o Show product
  - o Delete product
- ផ្នែក Order product
- ផ្នែក Show order product
- ផ្នែក Delete order product
- ផ្នែក Edit order product

**សំគាល់** : ដោយប្រើ Structure Array ក្នុងការផ្ទុកទិន្នន័យ

```
#include<stdio.h>
```

```
#include<conio.h>
```

```
#include<ctype.h>
```

```
#include<string.h>
```

```

#include<dos.h>

#define MAX 100

struct product{

    char pid[10];

    char name[20];

    int qty;

    float price;

};

struct order{

    char rid[10];

    char pid[10];

    int qty;

    float price;

};

int n=0,np=0,i,ii,j,b,k;

char ch;

char x[20],xpid[20];

int main_menu( );

void input( );

void output( );

int search_id( char * );

void delete_id( );

```

```

void edit_id( );

int menu_pro( );

product p[MAX];

order  r[MAX];

void main( ){

do{

    clrscr( );

    i=main_menu( );

    switch(i){

    case 0:

        do{

            clrscr( );

            ii=menu_pro( );

            switch( ii){

            case 0:

                do{

                    clrscr( );

                    printf( "Input product id: ");

                    fflush( stdin);

                    gets( p[np].pid);

                    printf( "Input product name: ");

```

```

gets( p[np].name );

printf( "Input product qty : " );

scanf( "%d",&p[np].qty );

printf( "Input product price: " );

scanf( "%f",&p[np].price );

np++;

printf( "Press key Enter to continue other back" );

}while( getch( )==13 );

break;

```

case 1:

```

clrscr( );

printf( "%10s%20s%10s%10s\n","RID","Pro_id","Qty","Price" );

for( j=0;j<np; j++ )

printf( "%10s%20s%10d%10.2f\n",p[j].pid,p[j].name,p[j].qty,p[j].price );

getch( );

break;

```

case 2:

```

clrscr( );

printf( "Enter product id to delete: " );

fflush( stdin );

gets( x );

b=0;

```

```

        for( j=0;j<np;j++)
        {
            if( strcmp( p[j].pid,x)==0 )
            {
                b=1;
                for( k=j; k<np; k++)
                p[k] = p[k+1];

                np--;
            }
        }

        if( b==0 )

            printf( "Delete product not completed " );

        else

            printf( "Delete completed" );

        getch( );

        break;

    }

}while( ii!=3);

break;

case 1:

    input( );

    break;

```



case 2:

output( );

break;

case 3:

textbackground( BLUE );

clrscr( );

printf( "Enter order id to Search : " );

scanf( "%s",&x );

b=search\_id( x );

if( b==0 ){

printf( "Order id not found" );

}

getch( );

break;

case 4:

delete\_id( );

break;

case 5:

edit\_id( );

break;

}

}while( i!=6 );

```

}

void input( ){

float k;

    textbackground( BLUE );

do{

again:

    clrscr( );

    gotoxy( 1,25 );

    printf( "Enter order id='exit' to stop " );

    gotoxy( 1,1 );

    printf( "Enter order id: " );

    fflush( stdin );

    gets( r[n].rid );

    if( strcmp( r[n].rid,"exit")==0 )

    {

        break;

    }

    printf( "Enter product id: " );

    gets( r[n].pid );

    b=0;

    for( j=0; j<np; j++ )

    {

```

```

        if( strcmp( p[j].pid,r[n].pid )==0 ){

            b=1;

            printf( "Enter order qty: " );

            scanf( "%d",&r[n].qty );

            printf( "Enter order price: " );

            scanf( "%f",&k );

            r[n].price=k;

            p[j].qty = p[j].qty-r[n].qty;

            n=n+1;

        }

    }

    if( b==0 )

    {

        gotoxy( 1,13);

        printf( "Product id not found try again!" );

        getch( ); goto again;

    }

    printf( "Press Enter continue other exit" );

}while( getch( )==13);

}

void output( ){

textbackground( BLUE );

```

```

clrscr( );

if( n<8 )
{
    printf( "%10s%20s%10s%10s\n", "RID", "Pro_id", "Qty", "Price" );

    for( i=0; i<n; i++ )
    {
        printf( "%10s%20s%10d%10.2f\n", r[i].rid, r[i].pid, r[i].qty, r[i].price );
    }

    getch( );
} else
{
    j=0;

    do{
        clrscr( );

        printf( "%10s%20s%10s%10s\n", "RID", "Pro_id", "Qty", "Price" );

        for( i=j; i<j+8; i++ )
        {
            printf( "%10s%20s%10d%10.2f\n", r[i].rid, r[i].pid, r[i].qty, r[i].price );
        }

        ch=getch( );

        if( tolower( ch )=='a' )
        {
            j--; if( j<0 ) j++;
        }
    } while( ch!='q' );
}

```

```

    }

    if( tolower( ch )=='z' )

    {

        j++; if( j+8>n ) j--;

    }

}while( ch!=27 );

}

}

int search_id( char *x ){

    b=0;

    for( i=0;i<n;i++ )

    {

        if( strcmp( r[i].rid,x )==0 ){

            if( b==0 )

                printf( "%10s%20s%10s%10s\n","RID","Pro_id","Qty","Price" );

            b=1;

            printf( "%10s%20s%10d%10.2f\n",r[i].rid,r[i].pid,r[i].qty,r[i].price );

        }

    }

    return b;

}

void delete_id( ){

```

```

textbackground( BLUE );

clrscr( );

printf( "Enter order id to delete: " );

fflush( stdin );

gets( x );

printf( "Enter product id: " );

gets( xpid );

b=0;

for( i=0;i<n;i++ )

{

    if( strcmp( r[i].rid,x )==0 && strcmp( r[i].pid,xpid )==0 )

    {

        b=1;

        for( k=0;k<np;k++ )

        {

            if( strcmp( p[k].pid,r[i].pid )==0 )

                p[k].qty=p[k].qty+r[i].qty;

        }

        for( j=i;j<n-1;j++ )

        {

            r[j]=r[j+1];

        }

    }

}

```

```

        n=n-1;

    }

}

if( b==0 )

    printf( "Delete not completed" );

else

    printf( "Delete completed" );

    getch( );

}

void edit_id( ){

    textbackground( BLUE );

    clrscr( );

    printf( "Enter order id to Edit: " );

    fflush( stdin );

    gets( x );

    printf( "Enter product id to Edit: " );

    gets( xpid );

    b=0;

    for( i=0;i<n;i++ )

    {

        if( strcmp( r[i].rid,x)==0 && strcmp( r[i].pid,xpid)==0 )

```

```

{

b=1;

for( k=0;k<np;k++ )

{

    if( strcmp( p[k].pid,r[i].pid )==0 ){

        p[k].qty=p[k].qty+r[i].qty;

        break;

    }

}

printf( "Enter new product id " );

scanf( "%s",&r[i].pid );

printf( "Enter new order qty: " );

scanf( "%d",&r[i].qty );

printf( "Enter new price: " );

scanf( "%f",&r[i].price );

p[k].qty=p[k].qty-r[i].qty;

}

}

if( b==0 )

    printf( "Edit not completed" );

else

    printf( "Edit Update completed" );

```



```

        getch( );
    }

    int main_menu( ){
do{

        textbackground( BLUE );

        clrscr( );

        textcolor( WHITE );

        textbackground( GREEN );

        gotoxy( 1,9+i);

        cprintf( "%30c",' ');

        gotoxy( 1,9);

        printf( "Check product in Stock");

        gotoxy( 1,10);

        printf( "Register information student");

        gotoxy( 1,11 );

        printf( "Show information student");

        gotoxy( 1,12);

        printf( "Search information student");

        gotoxy( 1,13);

        printf( "Delete information student");

        gotoxy( 1,14);

        printf( "Edit information student");

```

```

gotoxy( 1,15);

printf( "Exit program\n");

gotoxy( 1,25);

printf( "Use key to Scroll 'A' move 'UP' or 'Z' move 'Down'");

printf( " Enter='Select'");

ch=getch( );

switch( tolower( ch ) ){

case 'a':

    i=i-1;

    if( i<0 ) i=6;

    break;

case 'z':

    i=i+1;

    if( i>6 ) i=0;

}

}while( ch!=27 && ch!=13 );

if( ch==27 ) i=6;

return i;

}

int menu_pro( ){

do{

    textbackground( BLUE );

```

```

clrscr( );

textcolor( WHITE );

textbackground( GREEN );

gotoxy( 1,10+i);

cprintf( "%30c",' ');

gotoxy( 1,10);

printf( "Input product in stock" );

gotoxy( 1,11 );

printf( "Show product in stock" );

gotoxy( 1,12);

printf( "Delete product in stock" );

gotoxy( 1,13);

printf( "Exit program\n" );

gotoxy( 1,25);

printf( "Use key to Scroll 'A' move 'UP' or 'Z' move 'Down'" );

printf( " Enter='Select'" );

ch=getch( );

switch( tolower( ch ) ){

case 'a':

    i=i-1;

    if( i<0 ) i=3;

    break;

```

```
        case 'z':  
            i=i+1;  
            if( i>3 ) i=0;  
        }  
    }while( ch!=27 && ch!=13 );  
    if( ch==27 ) i=3;  
    return i;  
}
```

## មេរៀនទី១២

## របៀបបង្កើតនិងប្រើប្រាស់ Text File

### I. និយមន័យ Text File

Text File ជាការរក្សាទុកទិន្នន័យជាអថេរត្រួតត្រាក្នុងម៉ាស៊ីនជាទំរង់ Text ។ Text File គឺជា File ដែលយើងអាចបើកមើលទិន្នន័យដែលបានរក្សាទុកបានគ្រប់កម្មវិធីដែលជា Text Editor។ ខាងក្រោមជាការបង្ហាញនៃការប្រើ។

### II. ការប្រើប្រាស់ fopen

fopen ជាឃ្លាបញ្ជាប្រើសំរាប់បើក File ណាមួយពីក្នុង System File យកមកប្រើក្នុងកម្មវិធី C ។ខាងក្រោមជាទំរង់ទូទៅនៃការប្រើ

Syntax:

FILE \*fopen( const char \* Filename, const char \* Mode );

- Filename: ជាឈ្មោះ File ដែលត្រូវជ្រើសរើសយកមកប្រើ
- Mode : ជាប្រភេទនៃគោលបំណងដែលត្រូវបើក File យកមកប្រើ

| Sr.No. | Mode & ការពិពណ៌នា                                                                                                                               |
|--------|-------------------------------------------------------------------------------------------------------------------------------------------------|
| 1      | <b>r</b><br>គោលបំណងបើក File ដែលមានហើយយកមកអាន                                                                                                    |
| 2      | <b>w</b><br>គោលបំណងបើក File មកកាត់ត្រាទិន្នន័យថ្មីទុក បើ File មានទិន្នន័យហើយនិងលប់អាចាស់បង្កើតអាថ្នី បើ File មិនទាន់មាននិងបង្កើតថ្មីមួយ         |
| 3      | <b>a</b><br>គោលបំណងបើក File មកកាត់ត្រាទិន្នន័យទុកបន្ត បើ File មានទិន្នន័យហើយនិងបន្ថែមទិន្នន័យចូលបន្ត បើ File មិនទាន់មាននិងបង្កើតថ្មីមួយ         |
| 4      | <b>r+</b><br>គោលបំណងបើក File មកទាំងរក្សាទិន្នន័យនិងអានទិន្នន័យ                                                                                  |
| 5      | <b>w+</b><br>គោលបំណងបើក File មកទាំងរក្សាទិន្នន័យនិងអានទិន្នន័យបើ File មានទិន្នន័យហើយនិងលប់អាចាស់បង្កើតអាថ្នី បើ File មិនទាន់មាននិងបង្កើតថ្មីមួយ |
| 6      | <b>a+</b>                                                                                                                                       |

|  |                                                                                                                                     |
|--|-------------------------------------------------------------------------------------------------------------------------------------|
|  | គោលបំណងបើក File មកទាំងរក្សាទិន្នន័យនិងអានទិន្នន័យ បើ File មានទិន្នន័យហើយនិងបន្ថែមទិន្នន័យចូលបន្ត បើ File មិនទាន់មាននិងបង្កើតថ្មីមួយ |
|--|-------------------------------------------------------------------------------------------------------------------------------------|

III. **ការប្រើប្រាស់ fclose**

fclose ជាឃ្លាបញ្ជាសំរាប់បិទបញ្ចប់នៃការប្រើ File ណាមួយ។ខាងក្រោមជាទំរង់ទូទៅនៃការប្រើ

Syntax:

```
int fclose( FILE *fp );
```

IV. **ការប្រើប្រាស់អក្ខរកម្មទិន្នន័យ**

1. **ការប្រើ fputc**

fputc ជាឃ្លាប្រើសំរាប់ទុកទិន្នន័យម្តងម្កាលអក្សរចូលទៅក្នុង File។ ខាងក្រោមជាឧទាហរណ៍នៃការប្រើ

Syntax:

```
int fputc( int c, FILE *fp );
```

**ខាងក្រោមជាកម្មវិធីបង្ហាញនៃការប្រើ**

```
#include<stdio.h>
#include<conio.h>
char ch;
void main( ){
    FILE *fp;
    fp = fopen( "/tmp/test.txt", "w+" );
    clrscr( );
    printf( "Please Enter character: " );
    do{
        ch=getchar( );
        fputc( ch,fp);
    }while( ch!='#' );
}
```

2. **ការប្រើ fputs**

fputs ជាឃ្លាប្រើសំរាប់ទុកទិន្នន័យម្តងច្រើនអក្សរចូលទៅក្នុង File។ ខាងក្រោមជាឧទាហរណ៍នៃការប្រើ

Syntax:

```
int fputs( const char *s, FILE *fp );
```

## ខាងក្រោមជាកម្មវិធីបង្ហាញនៃការប្រើ

```
#include <stdio.h>

main( ) {
    FILE *fp;

    fp = fopen( "/tmp/test.txt", "w+" );
    fputs( "This is testing for fputs...\n", fp );
    fclose( fp );
}
```

### 3. ការប្រើ fprintf

fprintf ជាឃ្លាប្រើសំរាប់ទុកទិន្នន័យម្តងច្រើនអក្សរចូលទៅក្នុង File។ ខាងក្រោមជាឧទាហរណ៍នៃការប្រើ

Syntax:

```
void fprintf( FILE *fp, "Format_list", arg1, ... );
```

## ខាងក្រោមជាកម្មវិធីបង្ហាញនៃការប្រើ

```
#include <stdio.h>

main( ) {
    FILE *fp;

    fp = fopen( "/tmp/test.txt", "w+" );
    fprintf( fp, "This is testing for fprintf...\n" );
    fclose( fp );
}
```

## V. ការប្រើបញ្ចូលអានទិន្នន័យ

### 1. ការប្រើ fgetc

fgetc ជាឃ្លាបញ្ចូលប្រើសំរាប់អានទិន្នន័យម្តងមួយអក្សរចេញពីក្នុង File។ ខាងក្រោមជាឧទាហរណ៍នៃការប្រើ

Syntax:

```
int fgetc( FILE * fp );
```

## ខាងក្រោមជាកម្មវិធីបង្ហាញនៃការប្រើ

```
#include<stdio.h>
#include<conio.h>

char ch;
```

```

void main( ){
    clrscr( );
    FILE *fp;
    fp = fopen( "/tmp/test.txt", "r" );
    do{
        ch=fgetc( fp );
        printf( "%c",ch );
    }while( ch!=EOF );
    getch( );
}

```

## 2. ការប្រើ fgetc

fgetc ជាប្លង់បញ្ជាប្រើសំរាប់អានទិន្នន័យម្តងច្រើនអក្សរចេញពីក្នុង File។ ខាងក្រោមជាឧទាហរណ៍នៃការប្រើ

Syntax:

```
char *fgetc( char *buf, int n, FILE *fp );
```

### ខាងក្រោមជាកម្មវិធីបង្ហាញនៃការប្រើ

```

#include <stdio.h>

void main( ) {
    FILE *fp;
    char buff[255];
    fp = fopen( "/tmp/test.txt", "r" );
    fgetc( buff, 255, ( FILE * )fp );
    printf( "1 : %s\n", buff );
    fgetc( buff, 255, ( FILE * )fp );
    printf( "2: %s\n", buff );
    fgetc( buff, 255, ( FILE * )fp );
    printf( "3: %s\n", buff );
    fclose( fp );
}

```

## 3. ការប្រើ fscanf



fgets ជាបញ្ជាប្រើសំរាប់អានទិន្នន័យម្តងច្រើនអក្សរចេញពីក្នុង File។ ខាងក្រោមជាឧទាហរណ៍នៃការប្រើ

Syntax:

```
int fscanf( FILE *fp,"Format_control",&arg1,...);
```

### **ខាងក្រោមជាកម្មវិធីបង្ហាញនៃការប្រើ**

```
#include <stdio.h>
```

```
void main( ) {
```

```
    FILE *fp;
```

```
    char buff[255];
```

```
    fp = fopen( "/tmp/test.txt", "r" );
```

```
    fscanf( fp, "%s", buff );
```

```
    printf( "1 : %s\n", buff );
```

```
    fscanf( fp, "%s", buff );
```

```
    printf( "2: %s\n", buff );
```

```
    fscanf( fp, "%s", buff );
```

```
    printf( "3: %s\n", buff );
```

```
    fclose( fp );
```

```
}
```

## **VI. ការប្រើបញ្ជា rename**

rename ជាបញ្ជាប្រើសំរាប់ផ្លាស់ប្តូរឈ្មោះ File ណាមួយដែលមានក្នុង System ។ ខាងក្រោមជាទំរង់ទូទៅនៃការប្រើ

Syntax:

```
int rename( const char *old_name, const char *new_name );
```

Ex:

```
#include <stdio.h>
```

```
void main( ) {
```

```
    FILE *fp;
```

```
    fp = fopen( "/tmp/test.txt", "w" );
```

```
    rename( "/tem/test.txt", "/tem/mytest.txt" );
```

```
}
```

## **VII. ការប្រើបញ្ជា remove**

remove ជាបញ្ជាប្រើសំរាប់លុប File ណាមួយចេញពី System ។ខាងក្រោមជាទំរង់ទូទៅនៃការប្រើ  
Syntax:

```
int remove( const char *File_name );
```

Ex:

```
#include <stdio.h>

void main( ) {
    FILE *fp;
    fp = fopen( "/tmp/test.txt", "w" );
    remove( "/tmp/test.txt" );
}
```

#### VIII. ការប្រើ Search

ការពិរមៀនមុខៗយើងបានឃើញពីវិធីសាស្ត្រស្វែងរកទិន្នន័យនៅក្នុង Array, Structure រួចហើយចំណែក  
ទិន្នន័យដែលប្រើដោយ File ទុកក៏ត្រូវស្វែងរកបានដូចគ្នា។ខាងក្រោមជាឧទាហរណ៍នៃការប្រើ

```
#include<stdio.h>
#include<conio.h>
char room_num[20];
char room_type[20];
char room_price[10];
char room_status[20];
void main( )
{
    int n,i,b;
    char x_name[20];
    clrscr( );
    FILE *fp = fopen( "room.txt","w" );
    printf( "Enter number of room information: " );
    scanf( "%d",&n );
    for( i=1; i<=n; i++){
        printf( "Enter room number: " );
```

```

scanf( "%s",&room_num );
printf( "Enter room type: " );
scanf( "%s",&room_type );
printf( "Enter room price: " );
scanf( "%s",&room_price );
strcpy( room_status,"Free" );
fprintf( fp,"%s %s %s %s ",room_num,
                                room_type,
                                room_price,
                                room_status );

} fclose( fp );
printf( "Enter room type to search: " );
scanf( "%s",&x_name );
b=0;
fp=fopen( "room.txt","r" );
while( fscanf( fp,"%s%s%s%s",&room_num,
                                &room_type,
                                &room_price,
                                &room_status )==4 )
{
    if( strcmp( room_type,x_name )==0 )
    {
        b=1;
        printf( "%10s%10s%10s%10s\n",    room_num,
                                            room_type,
                                            room_price,
                                            room_status );
    }
}
if( b==0 )
    printf( "Room type not found" );

```

```

        getch( );
        fclose( fp );
    }

```

### ខាងក្រោមជាលទ្ធផលទី១របស់កម្មវិធីខាងលើ

Enter number room information: 2

Enter room number: R001

Enter room type: vip

Enter room price: 50

Enter room number: R002

Enter room type: Air\_room

Enter room price: 25

Enter room type to search: normal

Room type not found

### ខាងក្រោមជាលទ្ធផលទី២របស់កម្មវិធីខាងលើ

Enter number room information: 2

Enter room number: R001

Enter room type: vip

Enter room price: 50

Enter room number: R002

Enter room type: Air\_room

Enter room price: 25

Enter room type to search: vip

|      |     |    |      |
|------|-----|----|------|
| R001 | vip | 50 | Free |
|------|-----|----|------|

## IX. ការប្រើ Sort

ការពីរមេរៀនមុខៗយើងបានឃើញពីវិធីសាស្ត្រតម្រាបទិន្នន័យនៅក្នុង Array, Structure រួចហើយចំណែកទិន្នន័យដែរប្រើដោយ File ទុកក៏ត្រូវការតម្រាបដូចគ្នា។ខាងក្រោមជាឧទាហរណ៍នៃការប្រើ

```

#include<stdio.h>
#include<conio.h>
#include<string.h>

```

```

struct ROOM{
char room_num[20];
char room_type[20];
char room_price[10];
char room_status[20];
};

void main( ){
    ROOM room[5]={
        {"R001","normal_room","15","Free"},
        {"R002","normal_room","15","Free"},
        {"R003","vip_room","50","Free"},
        {"R004","air_room","25","Free"},
        {"R005","air_room","25","Free"}
    };

    ROOM R;
    int i,j,k;
    clrscr( );
    FILE *fp = fopen( "room.txt","w" );
    for( i=0; i<5; i++){
        fprintf( fp,"%s %s %s %s ",room[i].room_num,
                room[i].room_type,
                room[i].room_price,
                room[i].room_status );

    } fclose( fp );
    printf( "Information room before Soft( type):\n" );
    fp=fopen( "room.txt","r" );
    i=0;
    while( fscanf( fp,"%s%s%s%s",      &room[i].room_num,
                &room[i].room_type,
                &room[i].room_price,
                &room[i].room_status)==4 )
    {
        printf( "%10s%20s%10s%10s\n",room[i].room_num,

```

```

        room[i].room_type,
        room[i].room_price,
        room[i].room_status);

        i++;
    }
    fclose( fp );
    fp=fopen( "room.txt","w" );
    for( j=0;j<i;j++ )
    {
        for( k=j+1;k<i;k++ )
        {
            if( strcmp( room[j].room_type,room[k].room_type )>0 )
            {
                R=room[j];
                room[j]=room[k];
                room[k]=R;
            }
        }
        fprintf( fp,"%s %s %s %s ",room[j].room_num,
                room[j].room_type,
                room[j].room_price,
                room[j].room_status );
    }
    fclose( fp );

    printf( "Information room after Soft( type):\n" );
    fp=fopen( "room.txt","r" );
    i=0;
    while( fscanf( fp,"%s%s%s%s",      &R.room_num,
                &R.room_type,
                &R.room_price,
                &R.room_status )==4 )
    {

```

```

printf( "%10s%20s%10s%10s\n",R.room_num,
R.room_type,
R.room_price,
R.room_status );
}
getch( );
fclose( fp );
}

```

### ខាងក្រោមជាលទ្ធផលរបស់កម្មវិធីខាងលើ

Information room before Soft( type ):

|      |             |    |      |
|------|-------------|----|------|
| R001 | normal_room | 15 | Free |
| R002 | normal_room | 15 | Free |
| R003 | vip_room    | 50 | Free |
| R004 | air_room    | 25 | Free |
| R005 | air_room    | 25 | Free |

Information room after Soft( type ):

|      |             |    |      |
|------|-------------|----|------|
| R004 | air_room    | 25 | Free |
| R005 | air_room    | 25 | Free |
| R001 | normal_room | 15 | Free |
| R002 | normal_room | 15 | Free |
| R003 | vip_room    | 50 | Free |

### X. ការប្រើ Delete

ការពិរេកមុខៗយើងបានឃើញពីវិធីសាស្ត្រលុបទិន្នន័យនៅក្នុង Array, Structure រួចហើយចំណែកទិន្នន័យដែរប្រើដោយ File ទុកក៏ត្រូវការលុបដូចគ្នា។ខាងក្រោមជាឧទាហរណ៍នៃការប្រើ

```

#include<stdio.h>
#include<conio.h>
#include<string.h>
struct ROOM{
char room_num[20];
char room_type[20];
char room_price[10];

```

```

char room_status[20];
};
void main( ){
    ROOM room[5]={
        {"R001","normal_room","15","Free"},
        {"R002","normal_room","15","Free"},
        {"R003","vip_room","50","Free"},
        {"R004","air_room","25","Free"},
        {"R005","air_room","25","Free"}
    };

    ROOM R;
    int i,j,k;
    char x_num[20];
    clrscr( );
    FILE *fp1;
    FILE *fp = fopen( "room.txt","w" );
    for( i=0; i<5; i++){
        fprintf( fp,"%s %s %s %s ",room[i].room_num,
                room[i].room_type,
                room[i].room_price,
                room[i].room_status);
    } fclose( fp );
    printf( "Information room before delete:\n" );
    fp=fopen( "room.txt","r" );
    i=0;
    while( fscanf( fp,"%s%s%s%s",      &R.room_num,
                &R.room_type,
                &R.room_price,
                &R.room_status)==4 )
    {
        printf( "%10s%20s%10s%10s\n",R.room_num,
                R.room_type,
                R.room_price,

```



```

                                R.room_status);
    }
    fclose( fp );
    printf( "Enter room number to delete: " );
    scanf( "%s",&x_num );
    fp=fopen( "room.txt","r" );
    fp1=fopen( "temp.txt","w" );
    i=0;
    while( fscanf( fp,"%s%s%s%s",      &R.room_num,
                                &R.room_type,
                                &R.room_price,
                                &R.room_status )==4 )
    {
        if( strcmp( R.room_num,x_num ) !=0 )
        {
            fprintf( fp1,"%s %s %s %s ",R.room_num,
                                R.room_type,
                                R.room_price,
                                R.room_status );
        }else
            i=1;
    }
    fclose( fp );
    fclose( fp1 );
    remove( "room.txt" );
    rename( "temp.txt","room.txt" );
    if( i==0 )
        printf( "Delete not completed\n" );
    else
        printf( "Delete completed\n" );
    printf( "Information room after delete:\n" );
    fp=fopen( "room.txt","r" );
    i=0;

```

```

while( fscanf( fp,"%s%s%s%s",      &R.room_num,
                                &R.room_type,
                                &R.room_price,
                                &R.room_status)==4 )
{
    printf( "%10s%20s%10s%10s\n",R.room_num,
                                R.room_type,
                                R.room_price,
                                R.room_status );
}
fclose( fp );
getch( );
}

```

### **ខាងក្រោមជាលទ្ធផលទី១របស់កម្មវិធីខាងលើ**

Information room before delete:

|      |             |    |      |
|------|-------------|----|------|
| R001 | normal_room | 15 | Free |
| R002 | normal_room | 15 | Free |
| R003 | vip_room    | 50 | Free |
| R004 | air_room    | 25 | Free |
| R005 | air_room    | 25 | Free |

Enter room id to delete: R000

Delete not completed

Information room after delete:

|      |             |    |      |
|------|-------------|----|------|
| R004 | air_room    | 25 | Free |
| R005 | air_room    | 25 | Free |
| R001 | normal_room | 15 | Free |
| R002 | normal_room | 15 | Free |
| R003 | vip_room    | 50 | Free |

### **ខាងក្រោមជាលទ្ធផលទី២របស់កម្មវិធីខាងលើ**

Information room before delete:

|      |             |    |      |
|------|-------------|----|------|
| R001 | normal_room | 15 | Free |
|------|-------------|----|------|

|      |             |    |      |
|------|-------------|----|------|
| R002 | normal_room | 15 | Free |
| R003 | vip_room    | 50 | Free |
| R004 | air_room    | 25 | Free |
| R005 | air_room    | 25 | Free |

Enter room id to delete: R003

Delete completed

Information room after delete:

|      |             |    |      |
|------|-------------|----|------|
| R004 | air_room    | 25 | Free |
| R005 | air_room    | 25 | Free |
| R001 | normal_room | 15 | Free |
| R002 | normal_room | 15 | Free |

## XI. ការប្រើ Edit

ការពិនិត្យមុខៗយើងបានឃើញពីវិធីសាស្ត្រកែសម្រួលទិន្នន័យនៅក្នុង Array, Structure រួចហើយចំណែកទិន្នន័យដែលប្រើដោយ File ទុកក៏ត្រូវការកែសម្រួលដូចគ្នា។ខាងក្រោមជាឧទាហរណ៍នៃការប្រើ

```
#include<stdio.h>
#include<conio.h>
#include<string.h>
struct ROOM{
char room_num[20];
char room_type[20];
char room_price[10];
char room_status[20];
};
void main( ){
    ROOM room[5]={
        {"R001","normal_room","15","Free"},
        {"R002","normal_room","15","Free"},
        {"R003","vip_room","50","Free"},
        {"R004","air_room","25","Free"},
        {"R005","air_room","25","Free"}
```

```

        };

ROOM R;
int i,j,k;
char x_num[20];
clrscr( );
FILE *fp1;
FILE *fp = fopen( "room.txt","w" );
for( i=0; i<5; i++){
    fprintf( fp,"%s %s %s %s ",room[i].room_num,
                                room[i].room_type,
                                room[i].room_price,
                                room[i].room_status );

} fclose( fp );
printf( "Information room before Edit:\n" );
fp=fopen( "room.txt","r" );
i=0;
while( fscanf( fp,"%s%s%s%s",      &R.room_num,
                                &R.room_type,
                                &R.room_price,
                                &R.room_status )==4 )
{
    printf( "%10s%20s%10s%10s\n",R.room_num,
                                R.room_type,
                                R.room_price,
                                R.room_status );

}
fclose( fp );
printf( "Enter room number to Edit: " );
scanf( "%s",&x_num );
fp=fopen( "room.txt","r" );
fp1=fopen( "temp.txt","w" );
i=0;
while( fscanf( fp,"%s%s%s%s",      &R.room_num,

```

```

                                &R.room_type,
                                &R.room_price,
                                &R.room_status)==4)
{
    if( strcmp( R.room_num,x_num )==0)
    {
        i=1;
        printf( "Enter new room number: ");
        scanf( "%s",&R.room_num );
        printf( "Enter new room type: ");
        scanf( "%s",&R.room_type );
        printf( "Enter new room price");
        scanf( "%s",&R.room_price );
        printf( "Enter new room status: ");
        scanf( "%s",&R.room_status );
    }
    fprintf( fp1,"%s %s %s %s ",R.room_num,
                                R.room_type,
                                R.room_price,
                                R.room_status );
}
fclose( fp );
fclose( fp1 );
remove( "room.txt" );
rename( "temp.txt","room.txt" );
if( i==0)
    printf( "Edit not completed\n" );
else
    printf( "Edit update completed\n" );
printf( "Information room after edit:\n" );
fp=fopen( "room.txt","r" );
i=0;
while( fscanf( fp,"%s%s%s%s",    &R.room_num,

```

```

                                &R.room_type,
                                &R.room_price,
                                &R.room_status )==4 )
        {
            printf( "%10s%20s%10s%10s\n",R.room_num,
                    R.room_type,
                    R.room_price,
                    R.room_status );
        }
        fclose( fp );
        getch( );
    }
}

```

### ខាងក្រោមជាលទ្ធផលទី១របស់កម្មវិធីខាងលើ

Information room before edit:

|      |             |    |      |
|------|-------------|----|------|
| R001 | normal_room | 15 | Free |
| R002 | normal_room | 15 | Free |
| R003 | vip_room    | 50 | Free |
| R004 | air_room    | 25 | Free |
| R005 | air_room    | 25 | Free |

Enter room id to edit: R000

Edit not completed

Information room after delete:

|      |             |    |      |
|------|-------------|----|------|
| R004 | air_room    | 25 | Free |
| R005 | air_room    | 25 | Free |
| R001 | normal_room | 15 | Free |
| R002 | normal_room | 15 | Free |
| R003 | vip_room    | 50 | Free |

### ខាងក្រោមជាលទ្ធផលទី២របស់កម្មវិធីខាងលើ

Information room before edit:

|      |             |    |      |
|------|-------------|----|------|
| R001 | normal_room | 15 | Free |
| R002 | normal_room | 15 | Free |

|      |          |    |      |
|------|----------|----|------|
| R003 | vip_room | 50 | Free |
| R004 | air_room | 25 | Free |
| R005 | air_room | 25 | Free |

Enter room id to edit: R003

Enter new room number: R003

Enter new room type: special\_room

Enter new room price: 100

Enter new room status: Free

Edit update completed

Information room after edit:

|      |              |     |      |
|------|--------------|-----|------|
| R004 | air_room     | 25  | Free |
| R005 | air_room     | 25  | Free |
| R001 | normal_room  | 15  | Free |
| R002 | normal_room  | 15  | Free |
| R003 | special_room | 100 | Free |

## XII. លំហាត់ការគ្រប់គ្រងផ្នែកលក់ទំនិញក្នុងបារម្ភ (Update)

កម្មវិធីហាងលក់ទំនិញមានផ្នែកច្រើនក្នុងការគ្រប់គ្រងកម្មវិធីខាងក្រោមលើកយកតែផ្នែកលក់មកបង្ហាញមានការងារដូចជា

- ផ្នែក Stock product
  - o Input product
  - o Show product
  - o Delete product
- ផ្នែក Order product
- ផ្នែក Show order product
- ផ្នែក Delete order product
- ផ្នែក Edit order product

**សំគាល់** : ដោយប្រើ Text File ក្នុងការផ្ទុកទិន្នន័យ

```
#include<stdio.h>
```

```

#include<conio.h>

#include<ctype.h>

#include<string.h>

#include<dos.h>

#include<stdlib.h>

struct product{

    char pid[10];

    char name[20];

    char qty[10];

    char price[10];

};

struct order{

    char rid[10];

    char pid[10];

    char qty[10];

    char price[10];

};

int i,ii,j,b,k;

char ch;

char x[20],xpid[20];

int main_menu( );

void input( );

```



```

void output( );

int search_id( char * );

void delete_id( );

void edit_id( );

int menu_pro( );

product p;

order r;

FILE *fp, *fp1, *fp2, *fp3;

void main( ){

do{

    clrscr( );

    i=main_menu( );

    switch(i){

    case 0:

        do{

            clrscr( );

            ii=menu_pro( );

            switch( ii){

            case 0:

                do{

                    clrscr( );

                    printf( "Input product id: ");

```

```

fflush( stdin );

scanf( "%s",&p.pid );

printf( "Input product name: " );

scanf( "%s",&p.name );

printf( "Input product qty : " );

scanf( "%s",&p.qty );

printf( "Input product price: " );

scanf( "%s",&p.price );

fp=fopen( "Epro.txt","a" );

fprintf( fp,"%s %s %s %s ",p.pid,p.name,p.qty,p.price );

fclose( fp );

printf( "Press key Enter to continue other back" );

}while( getch( )==13 );

break;

```

case 1:

```

clrscr( );

printf( "%10s%20s%10s%10s\n","PID","Pro_name","Qty","Price" );

fp=fopen( "Epro.txt","r" );

if( fp==NULL )

{

    printf( "File not found" );

}
else

```

```

{
while( fscanf( fp,"%s%s%s%s",&p.pid,&p.name,&p.qty,&p.price )==4 )

    {

        printf( "%10s%20s%10s%10s\n",p.pid,p.name,p.qty,p.price );

    }

}

getch( );

fclose( fp );

break;

```

case 2:

```

clrscr( );

printf( "Enter product id to delete: " );

fflush( stdin );

gets( x );

b=0;

fp=fopen( "Epro.txt","r" );

fp1=fopen( "temp.txt","w" );

while( fscanf( fp,"%s%s%s%s",&p.pid,&p.name,&p.qty,&p.price )==4 )

{

    if( strcmp( p.pid,x )!=0 )

    {

        fprintf( fp1,"%s %s %s %s ",p.pid,p.name,p.qty,p.price );

    }else

```

```

        b=1;

    }

    fclose( fp1 );

    fclose( fp );

    remove( "Epro.txt" );

    rename( "temp.txt","Epro.txt" );

    if( b==0 )

        printf( "Delete product not completed " );

    else

        printf( "Delete completed" );

    getch( );

    break;

}

}while( ii!=3 );

break;

case 1:

    input( );

    break;

case 2:

    output( );

    break;

case 3:

```

```

        textbackground( BLUE );

        clrscr( );

        printf( "Enter order id to Search : " );

        scanf( "%S",&x );

        b=search_id( x );

        if( b==0 ){

            printf( "Order id not found" );

        }

        getch( );

        break;

    case 4:

        delete_id( );

        break;

    case 5:

        edit_id( );

        break;

    }

}while( i!=6 );

}

void input( ){

float k;

        textbackground( BLUE );

```

```

do{
again:

    clrscr( );

    gotoxy( 1,25 );

    printf( "Enter order id='exit' to stop " );

    gotoxy( 1,1 );

    printf( "Enter order id: " );

    scanf( "%s",&r.riid );

    if( strcmp( r.riid,"exit" )==0 )

    {

        break;

    }

    printf( "Enter product id: " );

    scanf( "%s",&r.pid );

    b=0;

    fp=fopen( "Epro.txt","r" );

    fp1=fopen( "temp.txt","w" );

    while( fscanf( fp,"%s%s%s%s",&p.pid,&p.name,&p.qty,&p.price )==4 )

    {

        if( strcmp( p.pid,r.pid )==0 ){

            b=1;

            printf( "Enter order qty: " );

```

```

scanf( "%s",&r.qty );

printf( "Enter order price: " );

scanf( "%s",&r.price );

itoa( atoi( p.qty )-atoi( r.qty ),p.qty,10 );

fp2=fopen( "Eorder.txt","a" );

fprintf( fp2,"%s %s %s %s ",r.rid,r.pid,r.qty,r.price );

fclose( fp2 );

}

fprintf( fp1,"%s %s %s %s ",p.pid,p.name,p.qty,p.price );

}

fclose( fp );

fclose( fp1 );

remove( "Epro.txt" );

rename( "temp.txt","Epro.txt" );

if( b==0 )

{

    gotoxy( 1,13 );

    printf( "Product id not found try again!" );

    getch( ); goto again;

}

printf( "Press Enter continue other exit" );

}while( getch( )==13 );

```

```

}

void output( ){

textbackground( BLUE );

clrscr( );

printf( "%10s%20s%10s%10s\n","RID","Pro_id","Qty","Price" );

fp=fopen( "Eorder.txt","r" );

if( fp==NULL )

{

printf( "File not found" );

}else

{

while( fscanf( fp,"%s%s%s%s",&r.rid,&r.pid,&r.qty,&r.price )==4 )

{

printf( "%10s%20s%10s%10s\n",r.rid,r.pid,r.qty,r.price );

}

}

getch( );

fclose( fp );

}

int search_id( char *x){

b=0;

fp=fopen( "Eorder.txt","r" );

```



```

if( fp==NULL )

{

    printf( "File not found" );

}

else

{

    while( fscanf( fp,"%s%s%s%s",&r.rid,&r.pid,&r.qty,&r.price )==4 )

    {

        if( strcmp( r.rid,x)==0 ){

            if( b==0 )

                printf( "%10s%20s%10s%10s\n","RID","Pro_id","Qty","Price" );

            b=1;

            printf( "%10s%20s%10s%10s\n",r.rid,r.pid,r.qty,r.price );

        }

    }

}

fclose( fp );

return b;

}

void delete_id( ){

    textbackground( BLUE );

    clrscr( );

```

```

printf( "Enter order id to delete: " );

scanf( "%s",&x );

printf( "Enter product id: " );

scanf( "%s",&xp );

b=0;

fp=fopen( "Eorder.txt","r" );

fp1=fopen( "temp.txt","w" );

while( fscanf( fp,"%s%s%s%s",&r. rid,&r. pid,&r. qty,&r. price )==4 )

{

    if( strcmp( r. rid,x)==0 && strcmp( r. pid,xp )==0 )

    {

        b=1;

        fp2=fopen( "Epro.txt","r" );

        fp3=fopen( "temp1.txt","w" );

        while( fscanf( fp2,"%s%s%s%s",&p. pid,&p. name,&p. qty,&p. price )==4 )

        {

            if( strcmp( p. pid,r. pid )==0 )

            {

                itoa( atoi( p. qty )+atoi( r. qty ),p. qty,10 );

            }

            fprintf( fp3,"%s %s %s %s ",p. pid,p. name,p. qty,p. price );

        }

    }

}

```

```

        fclose( fp2 );

        fclose( fp3 );

        remove( "Epro.txt" );

        rename( "temp1.txt", "Epro.txt" );

    }else

    {

        fprintf( fp1, "%s %s %s %s ", r.rid, r.pid, r.qty, r.price );

    }

}

fclose( fp );

fclose( fp1 );

remove( "Eorder.txt" );

rename( "temp.txt", "Eorder.txt" );

if( b==0 )

    printf( "Delete not completed" );

else

    printf( "Delete completed" );

getch( );

}

void edit_id( ){

    textbackground( BLUE );

    clrscr( );

```

```

printf( "Enter Order id to Edit: " );

scanf( "%s",&x );

printf( "Enter product id to Edit: " );

scanf( "%s",&xp );

b=0;

fp=fopen( "Eorder.txt","r" );

fp1=fopen( "temp.txt","w" );

while( fscanf( fp,"%s%s%s%s",&r. rid,&r. pid,&r. qty,&r. price )==4 )

{

    if( strcmp( r. rid,x)==0 && strcmp( r. pid,xp )==0 )

    {

        b=1;

        fp2=fopen( "Epro.txt","r" );

        fp3=fopen( "temp1.txt","w" );

        while( fscanf( fp2,"%s%s%s%s",&p. pid,&p. name,&p. qty,&p. price )==4 )

        {

            if( strcmp( p. pid,r. pid )==0 )

            {

                itoa( atoi( p. qty )+atoi( r. qty ),p. qty,10 );

            }

            fprintf( fp3,"%s %s %s %s ",p. pid,p. name,p. qty,p. price );

        }

    }

}

```

```

fclose( fp2 );

fclose( fp3 );

remove( "Epro.txt" );

rename( "temp1.txt","Epro.txt" );

printf( "Enter new product id " );

scanf( "%s",&r.pid );

printf( "Enter new order qty: " );

scanf( "%s",&r.qty );

printf( "Enter new price: " );

scanf( "%s",&r.price );

fp2=fopen( "Epro.txt","r" );

fp3=fopen( "temp1.txt","w" );

while( fscanf( fp2,"%s%s%s%s",&p.pid,&p.name,&p.qty,&p.price )==4 )

{

    if( strcmp( p.pid,r.pid )==0 )

    {

        itoa( atoi( p.qty )-atoi( r.qty ),p.qty,10 );

    }

    fprintf( fp3,"%s %s %s %s ",p.pid,p.name,p.qty,p.price );

}

fclose( fp2 );

fclose( fp3 );

```

```

        remove( "Epro.txt" );

        rename( "temp1.txt","Epro.txt" );

    }

    fprintf( fp1,"%s %s %s %s ",r.rid,r.pid,r.qty,r.price );

}

fclose( fp );

fclose( fp1 );

remove( "Eorder.txt" );

rename( "temp.txt","Eorder.txt" );

if( b==0 )

    printf( "Edit not completed" );

else

    printf( "Edit Update completed" );

    getch( );

}

int main_menu( ){

do{

    textbackground( BLUE );

    clrscr( );

    textcolor( WHITE );

    textbackground( GREEN );

    gotoxy( 1,9+i);

```

```

cprintf( "%30c", ' ');

gotoxy( 1,9);

printf( "Check product in Stock" );

gotoxy( 1,10);

printf( "Register information student" );

gotoxy( 1,11);

printf( "Show information student" );

gotoxy( 1,12);

printf( "Search information student" );

gotoxy( 1,13);

printf( "Delete information student" );

gotoxy( 1,14);

printf( "Edit information student" );

gotoxy( 1,15);

printf( "Exit program\n" );

gotoxy( 1,25);

printf( "Use key to Scroll 'A' move 'UP' or 'Z' move 'Down'" );

printf( " Enter='Select'" );

ch=getch( );

switch( tolower( ch ) ){

case 'a':

    i=i-1;

```

```

        if( i<0 ) i=6;

        break;

    case 'z':

        i=i+1;

        if( i>6 ) i=0;

    }

}while( ch!=27 && ch!=13 );

if( ch==27 ) i=6;

return i;

}

int menu_pro( ){

do{

    textbackground( BLUE );

    clrscr( );

    textcolor( WHITE );

    textbackground( GREEN );

    gotoxy( 1,10+i);

    cprintf( "%30c",' ');

    gotoxy( 1,10 );

    printf( "Input product in stock" );

    gotoxy( 1,11 );

    printf( "Show product in stock" );

```



```

gotoxy( 1,12 );

printf( "Delete product in stock" );

gotoxy( 1,13 );

printf( "Exit program\n" );

gotoxy( 1,25 );

printf( "Use key to Scroll 'A' move 'UP' or 'Z' move 'Down'" );

printf( " Enter='Select'" );

ch=getch( );

switch( tolower( ch ) ){

case 'a':

    i=i-1;

    if( i<0 ) i=3;

    break;

case 'z':

    i=i+1;

    if( i>3 ) i=0;

}

}while( ch!=27 && ch!=13 );

if( ch==27 ) i=3;

return i;

}

```

## មេរៀនទី១៣

## របៀបបង្កើតនិងប្រើប្រាស់ Binary File

### I. និយមន័យ Binary File

Binary File ជាការរក្សាទុកទិន្នន័យជាអថេរក្នុងម៉ាស៊ីនជាទំរង់ Binary ។ Binary File គឺជា File ដែរយើងមិនអាចបើកមើលទិន្នន័យដែរយើងបានរក្សាទុកក្នុងកម្មវិធីណាមួយបានទេ ពេលណាយើងត្រូវការយកទិន្នន័យមកប្រើវិញទាន់តែប្រើកម្មវិធីទើបបាន។ ខាងក្រោមជាការបង្ហាញនៃការប្រើ

### II. ការប្រើប្រាស់ fopen

fopen ជាប្លង់បញ្ជាប្រើសំរាប់បើក File ណាមួយពីក្នុង System File យកមកប្រើក្នុងកម្មវិធី C ។ខាងក្រោមជាទំរង់ទូទៅនៃការប្រើ

Syntax:

```
FILE *fopen( const char * Filename, const char * Mode );
```

- Filename : ជាឈ្មោះ File ដែរនិងត្រូវជ្រើសរើសយកមកប្រើ

- Mode : ជាប្រភេទនៃគោលបំណងដែរត្រូវបើក File យកមកប្រើ

| Sr.No. | Mode & ការពិពណ៌នា                                                                                                                       |
|--------|-----------------------------------------------------------------------------------------------------------------------------------------|
| 1      | <b>rb</b><br>គោលបំណងបើក File ដែរមានហើយយកមកអាន                                                                                           |
| 2      | <b>wb</b><br>គោលបំណងបើក File មកកត់ត្រាទិន្នន័យថ្មីទុក បើ File មានទិន្នន័យហើយនិងលប់អាចាស់បង្កើតថ្មី បើ File មិនទាន់មាននិងបង្កើតថ្មីមួយ   |
| 3      | <b>ab</b><br>គោលបំណងបើក File មកកត់ត្រាទិន្នន័យទុកបន្ត បើ File មានទិន្នន័យហើយនិងបន្ថែមទិន្នន័យចូលបន្ត បើ File មិនទាន់មាននិងបង្កើតថ្មីមួយ |
| 4      | <b>rb+, r+b</b>                                                                                                                         |

|   |                                                                                                                                                            |
|---|------------------------------------------------------------------------------------------------------------------------------------------------------------|
|   | គោលបំណងបើក File មកទាំងរក្សាទិន្នន័យនិងអានទិន្នន័យ                                                                                                          |
| 5 | <b>wb+, w+b</b><br><br>គោលបំណងបើក File មកទាំងរក្សាទិន្នន័យនិងអានទិន្នន័យបើ File មានទិន្នន័យហើយនិងលប់អាចាស់បង្កើតថ្មី បើ File មិនទាន់មាននិងបង្កើតថ្មីមួយ    |
| 6 | <b>ab+, a+b</b><br><br>គោលបំណងបើក File មកទាំងរក្សាទិន្នន័យនិងអានទិន្នន័យ បើ File មានទិន្នន័យហើយនិងបន្ថែមទិន្នន័យចូលបន្ត បើ File មិនទាន់មាននិងបង្កើតថ្មីមួយ |

### III. ការប្រើប្រាស់ fclose

fclose ជាប្លង់បញ្ជាសំរាប់បិទបញ្ចប់នៃការប្រើ File ណាមួយ។ខាងក្រោមជាទំរង់ទូទៅនៃការប្រើ

Syntax:

```
int fclose( FILE *fp );
```

### IV. ការប្រើប្រាស់ fwrite

fwrite ជាប្លង់បញ្ជាប្រើសំរាប់កត់ត្រាទិន្នន័យទុកក្នុង File ជាទំរង់ Binary ។ខាងក្រោមជាទំរង់ទូទៅនៃការប្រើប្លង់បញ្ជា fwrite

Syntax:

```
size_t fwrite( void *ptr, size_of_elements, number_of_elements, FILE *a_file );
```

- ptr : ជាអ្នកដៃផ្ទុកទិន្នន័យត្រូវកត់ត្រាទុកចូលទៅក្នុង File
- Size : ជាប្រវែងនៃចំនួនទិន្នន័យ
- Number : ជាចំនួនទិន្នន័យដែលត្រូវកត់ត្រា
- A\_file : ជា File ដែលចាំទទួលទិន្នន័យ

### ខាងក្រោមជាអម្បិចបង្ហាញនៃការប្រើ

```
#include <stdio.h>
```

```
struct Person
```

```
{
```

```
    char name[20];
```

```

        char sex[7];
        int age;

};

Person p = { "Dara", "Male", 20 };

main( ) {
    FILE *fp;
    fp = fopen( "/tmp/test.bin", "wb+" );
    fwrite( &p, sizeof( Person ), 1, fp );
    fclose( fp );
}

```

#### V. ការប្រើប្រាស់ fread

fread ជាប្លង់បញ្ជាប្រើសំរាប់អានទិន្នន័យដែលទុកក្នុង File ជាទំរង់ Binary មកប្រើវិញ។ ខាងក្រោមជាទំរង់ទូទៅនៃការប្រើប្លង់បញ្ជា fread

Syntax:

```
size_t fread( void *ptr, size_of_elements, number_of_elements, FILE *a_file );
```

- ptr : ជាអ្នកដែលផ្ទុកទិន្នន័យដែលត្រូវអានចេញពីក្នុង File មក
- Size : ជាប្រវែងនៃចំនួនទិន្នន័យ
- Number : ជាចំនួនទិន្នន័យដែលត្រូវអានចេញ
- A\_file : ជា File ដែលត្រូវអានទិន្នន័យចេញ

#### ខាងក្រោមជាអម្បិចបង្ហាញនៃការប្រើ

```
#include <stdio.h>
```

```
struct Person
```

```
{
    char name[20];
    char sex[7];
    int age;

};

```

```

    Person p;

main( ) {
    FILE *fp;
    fp = fopen( "/tmp/test.bin", "rb+" );
    fread( &p, sizeof( Person ), 1, fp );
    fclose( fp );
}

```

## VI. ការប្រើប្រាស់ `ftell`

`ftell` ជាបញ្ជីប្រើសំរាប់ប្រាប់ទំហំទិន្នន័យដែលផ្ទុកក្នុង File ជា byte ។ ខាងក្រោមជាទំនាក់ទំនងនៃការប្រើប្រាស់បញ្ជី `ftell`

Syntax:

```
long ftell( FILE *fp );
```

### ខាងក្រោមជាកម្មវិធីបង្ហាញនៃការប្រើ

```

#include <stdio.h>
#include <conio.h>

struct Person{
    char name[20];
    char sex[7];
    int age;
};

Person p = { "Dara", "male", 20 };

int main( void )
{
    FILE *stream;
    stream = fopen( "MYFILE.TXT", "wb+" );
    clrscr( );
    fwrite( &p, sizeof( Person ), 1, stream );
    printf( "The file pointer is at byte %ld\n", ftell( stream ) );
    getch( );
}

```

```

fclose( stream );

return 0;

}

```

### **ខាងក្រោមជាលទ្ធផលរបស់កម្មវិធីបង្ហាញនៃការប្រើខាងលើ**

The file pointer is at byte 29

## **VII. ការប្រើបញ្ជា rewind**

rewind ជាបញ្ជាប្រើសំរាប់កំណត់ទីតាំងរបស់ File pointer អោយមកនៅទីតាំងដំបូងនៃទិន្នន័យ។ ខាងក្រោមជាទំរង់ទូទៅនៃការប្រើរបស់បញ្ជា rewind

Syntax:

```
void rewind( FILE *fp );
```

### **ខាងក្រោមជាកម្មវិធីបង្ហាញនៃការប្រើ**

```

#include <stdio.h>
#include <conio.h>

struct Person{
    char name[20];
    char sex[7];
    int age;
};

Person p[2] = {
    {"Dara", "male", 20},
    {"Bopha", "Female", 20}
};

int main( void )
{
    FILE *stream;
    Person p;
    stream = fopen( "MYFILE.bin", "wb+" );
    clrscr( );
    fwrite( p[0], sizeof( Person ), 1, stream );
    fwrite( p[1], sizeof( Person ), 1, stream );
}

```

```

rewind( stream );

fread( &p, sizeof( Person ),1, stream );

printf( "%20s%10s%10d",p.name,p.sex,p.age );

getch( );

fclose( stream );

return 0;

}

```

### ខាងក្រោមជាលទ្ធផលរបស់កម្មវិធីបង្ហាញនៃការប្រើខាងលើ

```
Dara          male          20
```

## VIII. ការប្រើបញ្ជា fseek

fseek ជាបញ្ជាប្រើសំរាប់កំណត់ទីតាំងរបស់ File pointer អោយទៅទីតាំងនៃទិន្នន័យណាមួយក៏បាន។ ខាងក្រោមជាទំរង់ទូទៅនៃការប្រើរបស់បញ្ជា fseek

Syntax:

- ```
int      fseek( FILE *Fp, long Offset, int SEEK_MODE );
```
- Fp : ជា File ដែរផ្ទុកទិន្នន័យ
  - Offset : ជាទីតាំងដែរត្រូវបញ្ជាទៅ (គិតជា byte របស់ទិន្នន័យ)
  - SEEK\_MODE : ជាទីតាំងប្រៀបដែរត្រូវកំណត់ទៅមានបី
    - SEEK\_SET : ប្រៀបចាប់ពីទីតាំងដំបូងនៃទិន្នន័យ
    - SEEK\_CUR : ប្រៀបចាប់ពីទីតាំងនៃទិន្នន័យដែរកំពុងស្ថិតនៅ
    - SEEK\_END : ប្រៀបចាប់ពីទីតាំងចុងក្រោយនៃទិន្នន័យ

### ខាងក្រោមជាកម្មវិធីបង្ហាញនៃការប្រើ

```

#include <stdio.h>

#include <conio.h>

struct Person{
    char name[20];
    char sex[7];
    int age;
};

Person p[4] = {

```

```

        {"Dara", "male", 20},
        {"Bopha", "Female", 20},
        {"Thyda", "Female", 18},
        {"Kunthea", "male", 19}
    };

int main( void )
{
    FILE *stream;
    Person p;
    stream = fopen( "MYFILE.bin", "wb+" );
    clrscr( );
    fwrite( p[0],sizeof( Person ),1,stream );
    fwrite( p[1],sizeof( Person ),1,stream );
    fwrite( p[2],sizeof( Person ),1,stream );
    fwrite( p[3],sizeof( Person ),1,stream );
    fseek( stream,0,SEEK_SET );
    fread( &p, sizeof( Person ),1, stream );
    printf( "%20s%10s%10d",p.name,p.sex,p.age );
    fseek( stream, sizeof( Person ),SEEK_CUR );
    fread( &p, sizeof( Person ),1, stream );
    printf( "%20s%10s%10d",p.name,p.sex,p.age );
    fseek( stream, -3 * sizeof( Person ),SEEK_END );
    fread( &p, sizeof( Person ),1, stream );
    printf( "%20s%10s%10d",p.name,p.sex,p.age );
    fseek( stream, -1 * sizeof( Person ),SEEK_END );
    fread( &p, sizeof( Person ),1, stream );
    printf( "%20s%10s%10d",p.name,p.sex,p.age );
    getch( );
    fclose( stream );
    return 0;
}

```



}

### ខាងក្រោមជាលទ្ធផលរបស់កម្មវិធីបង្ហាញនៃការប្រើខាងលើ

|         |        |    |
|---------|--------|----|
| Dara    | male   | 20 |
| Thyda   | Female | 18 |
| Bopha   | Female | 20 |
| Kunthea | male   | 19 |

#### IX. ការប្រើ Search

ការពិរមៀនមុខៗយើងបានឃើញពីវិធីសាស្ត្រស្វែងរកទិន្នន័យនៅក្នុង Text file រួចហើយ។ ខាងក្រោមជាឧទាហរណ៍នៃការប្រើ Search នៅក្នុង Binary File វិញម្តង

```
#include<stdio.h>
#include<conio.h>
struct ROOM{
char room_num[20];
char room_type[20];
char room_price[10];
char room_status[20];
};
void main( )
{
    int n,i,b;
    char x_name[20];
    ROOM R;
    clrscr( );
    FILE *fp = fopen( "room.txt","wb" );
    printf( "Enter number of room information: " );
    scanf( "%d",&n );
    for( i=1; i<=n; i++){
```

```

printf( "Enter room number: " );
scanf( "%s",&R.room_num );
printf( "Enter room type: " );
scanf( "%s",&R.room_type );
printf( "Enter room price: " );
scanf( "%s",&R.room_price );
strcpy( R.room_status,"Free" );
fwrite( &p, sizeof( ROOM ), 1, fp );
} fclose( fp );
printf( "Enter room type to search: " );
scanf( "%s",&x_name );
b=0;
fp=fopen( "room.txt","rb" );
while( fread( &p, sizeof( ROOM ), 1, fp)==1 )
{
    if( strcmp( R.room_type,x_name)==0 )
    {
        b=1;
        printf( "%10s%10s%10s%10s\n",    R.room_num,
                                           R.room_type,
                                           R.room_price,
                                           R.room_status );
    }
}
if( b==0 )
    printf( "Room type not found" );
getch( );
fclose( fp );
}

```

**ខាងក្រោមជាលទ្ធផលទី១របស់កម្មវិធីខាងលើ**

Enter number room information: 2

Enter room number: R001  
 Enter room type: vip  
 Enter room price: 50  
 Enter room number: R002  
 Enter room type: Air\_room  
 Enter room price: 25  
 Enter room type to search: normal  
 Room type not found

### **ខាងក្រោមជាលទ្ធផលទី២របស់កម្មវិធីខាងលើ**

Enter number room information: 2  
 Enter room number: R001  
 Enter room type: vip  
 Enter room price: 50  
 Enter room number: R002  
 Enter room type: Air\_room  
 Enter room price: 25  
 Enter room type to search: vip

|      |     |    |      |
|------|-----|----|------|
| R001 | vip | 50 | Free |
|------|-----|----|------|

### **X. ការប្រើ Sort**

ការពីរមេរៀនមុខៗយើងបានឃើញពីវិធីសាស្ត្រតម្រៀបទិន្នន័យនៅក្នុង Text file រួចហើយ។ ខាងក្រោមជាឧទាហរណ៍នៃការប្រើ Sort នៅក្នុង Binary File វិញម្តង

```
#include<stdio.h>

#include<conio.h>

#include<string.h>

struct ROOM{

char room_num[20];

char room_type[20];
```

```

char room_price[10];

char room_status[20];

};

void main( ){

    ROOM room[5]={

        {"R001","normal_room","15","Free"},

        {"R002","normal_room","15","Free"},

        {"R003","vip_room","50","Free"},

        {"R004","air_room","25","Free"},

        {"R005","air_room","25","Free"}

    };

    ROOM R,R1,R2;

    int i,j,k,n;

    clrscr( );

    FILE *fp1, *fp2;

    FILE *fp = fopen( "room.txt","wb" );

    for( i=0; i<5; i++ )

    {

        fwrite( &room[i],sizeof( ROOM ),1,fp );

    }

    fclose( fp );

    printf( "Information room before Soft( type ):\n" );

```

```

fp=fopen( "room.txt","rb" );

while( fread( &R,sizeof( ROOM ),1,fp )==1 )

{

    printf( "%10s%20s%10s%10s\n",R.room_num,

                                                    R.room_type,

                                                    R.room_price,

                                                    R.room_status );

}

fclose( fp );

fp=fopen( "room.txt","rb+" );

fseek( fp,0,SEEK_END );

n=ftell( fp );

rewind( fp );

for( i=0;i<n;i++ )

{

    fseek( fp,i * sizeof( ROOM ),SEEK_SET );

    fread( &R,sizeof( ROOM ),1,fp );

    for( j=i+1;j<n;j++ )

    {

        fseek( fp,j * sizeof( ROOM ),SEEK_SET );

        fread( &R1,sizeof( ROOM ),1,fp );

        if( strcmp( R.room_type,R1.room_type )>0 )

```

```

{

fp1=fopen( "room.txt","rb" );

fp2=fopen( "temp1.txt","wb" );

    for( k=0;k<n;k++ )

    {

        fread( &R2,sizeof( ROOM ),1,fp1 );

        if( k==i )

        {

            R2 = R1;

        }

        else if( k==j )

        {

            R2 = R;

        }

        fwrite( &R2,sizeof( ROOM ),1,fp2 );

    }

fclose( fp1 );

fclose( fp2 );

remove( "room.txt" );

rename( "temp1.txt","room.txt" );

fp=fopen( "room.txt","rb+" );

R=R1;

```

```

        }

    }

}

fclose( fp );

printf( "Information room after Soft( type ):\n" );

getch( );

fclose( fp );

}

```

### ខាងក្រោមជាលទ្ធផលរបស់កម្មវិធីខាងលើ

Information room before Soft( type ):

|      |             |    |      |
|------|-------------|----|------|
| R001 | normal_room | 15 | Free |
| R002 | normal_room | 15 | Free |
| R003 | vip_room    | 50 | Free |
| R004 | air_room    | 25 | Free |
| R005 | air_room    | 25 | Free |

Information room after Soft( type ):

|      |             |    |      |
|------|-------------|----|------|
| R004 | air_room    | 25 | Free |
| R005 | air_room    | 25 | Free |
| R001 | normal_room | 15 | Free |
| R002 | normal_room | 15 | Free |
| R003 | vip_room    | 50 | Free |

### XI. ការប្រើ Delete

ការពិរមៀនមុខៗយើងបានឃើញពីវិធីសាស្ត្រលុបទិន្នន័យនៅក្នុង Text file រួចហើយ។ ខាងក្រោមជាឧទាហរណ៍នៃការប្រើ Delete នៅក្នុង Binary File វិញម្តង

```

#include<stdio.h>

#include<conio.h>

#include<string.h>

struct ROOM{

char room_num[20];

char room_type[20];

char room_price[10];

char room_status[20];

};

void main( ){

    ROOM room[5]={

        {"R001","normal_room","15","Free"},

        {"R002","normal_room","15","Free"},

        {"R003","vip_room","50","Free"},

        {"R004","air_room","25","Free"},

        {"R005","air_room","25","Free"}

    };

    ROOM R;

    int i,j,k;

    char x_num[20];

    clrscr( );

    FILE *fp1;

```



```

FILE *fp = fopen( "room.txt","wb" );

for( i=0; i<5; i++ )

{

fwrite( &room[i],sizeof( ROOM ),1,fp );

}

fclose( fp );

printf( "Information room before delete:\n" );

fp=fopen( "room.txt","rb" );

i=0;

while( fread( &R,sizeof( ROOM ),1,fp )==1 )

{

    printf( "%10s%20s%10s%10s\n",R.room_num,

                                                    R.room_type,

                                                    R.room_price,

                                                    R.room_status );

}

fclose( fp );

printf( "Enter room number to delete: " );

scanf( "%s",&x_num );

fp=fopen( "room.txt","rb" );

fp1=fopen( "temp.txt","wb" );

i=0;

```

```

while( fread( &R,sizeof( ROOM ),1,fp)==1 )
{
    if( strcmp( R.room_num,x_num )!=0 )
    {
        fwrite( &R,sizeof( ROOM ),1,fp1 );
    }else
        i=1;
}

fclose( fp );

fclose( fp1 );

remove( "room.txt" );

rename( "temp.txt","room.txt" );

if( i==0 )

    printf( "Delete not completed\n" );

else

    printf( "Delete completed\n" );

printf( "Information room after delete:\n" );

fp=fopen( "room.txt","rb" );

i=0;

while( fread( &R,sizeof( ROOM ),1,fp)==1 )
{

    printf( "%10s%20s%10s%10s\n",R.room_num,

```

```

        R.room_type,

        R.room_price,

        R.room_status );

    }

    fclose( fp );

    getch( );

}

```

### **ខាងក្រោមជាលទ្ធផលទី១របស់កម្មវិធីខាងលើ**

Information room before delete:

|      |             |    |      |
|------|-------------|----|------|
| R001 | normal_room | 15 | Free |
| R002 | normal_room | 15 | Free |
| R003 | vip_room    | 50 | Free |
| R004 | air_room    | 25 | Free |
| R005 | air_room    | 25 | Free |

Enter room id to delete: R000

Delete not completed

Information room after delete:

|      |             |    |      |
|------|-------------|----|------|
| R004 | air_room    | 25 | Free |
| R005 | air_room    | 25 | Free |
| R001 | normal_room | 15 | Free |
| R002 | normal_room | 15 | Free |
| R003 | vip_room    | 50 | Free |

## ខាងក្រោមជាលទ្ធផលទី២របស់កម្មវិធីខាងលើ

Information room before delete:

|      |             |    |      |
|------|-------------|----|------|
| R001 | normal_room | 15 | Free |
| R002 | normal_room | 15 | Free |
| R003 | vip_room    | 50 | Free |
| R004 | air_room    | 25 | Free |
| R005 | air_room    | 25 | Free |

Enter room id to delete: R003

Delete completed

Information room after delete:

|      |             |    |      |
|------|-------------|----|------|
| R004 | air_room    | 25 | Free |
| R005 | air_room    | 25 | Free |
| R001 | normal_room | 15 | Free |
| R002 | normal_room | 15 | Free |

## XII. ការប្រើ Edit

ការពីរមេរៀនមុខៗយើងបានឃើញពីវិធីសាស្ត្រកែសម្រួលទិន្នន័យនៅក្នុង Text file រួចហើយ។ ខាងក្រោមជាឧទាហរណ៍នៃការប្រើ Edit នៅក្នុង Binary File វិញម្តង

```
#include<stdio.h>
```

```
#include<conio.h>
```

```
#include<string.h>
```

```
struct ROOM{
```

```
char room_num[20];
```

```

char room_type[20];

char room_price[10];

char room_status[20];

};

void main( ){

    ROOM room[5]={

        {"R001","normal_room","15","Free"},

        {"R002","normal_room","15","Free"},

        {"R003","vip_room","50","Free"},

        {"R004","air_room","25","Free"},

        {"R005","air_room","25","Free"}

    };

    ROOM R;

    int i,j,k;

    char x_num[20];

    clrscr( );

    FILE *fp1;

    FILE *fp = fopen( "room.txt","wb" );

    for( i=0; i<5; i++ )

    {

        fwrite( &room[i],sizeof( ROOM ),1,fp );

    }

```

```

fclose( fp );

printf( "Information room before edit:\n" );

fp=fopen( "room.txt","rb" );

i=0;

while( fread( &R,sizeof( ROOM ),1,fp)==1 )

{

    printf( "%10s%20s%10s%10s\n",R.room_num,

                                                    R.room_type,

                                                    R.room_price,

                                                    R.room_status );

}

fclose( fp );

printf( "Enter room number to edit: " );

scanf( "%s",&x_num );

fp=fopen( "room.txt","rb" );

fp1=fopen( "temp.txt","wb" );

i=0;

while( fread( &R,sizeof( ROOM ),1,fp)==1 )

{

    if( strcmp( R.room_num,x_num )==0 )

    {

        i=1;

        printf( "Enter new room number: " );

```

```

        scanf( "%s",&R.room_num );
        printf( "Enter new room type: " );
        scanf( "%s",&R.room_type );
        printf( "Enter new room price" );
        scanf( "%s",&R.room_price );
        printf( "Enter new room status: " );
        scanf( "%s",&R.room_status );

    }

    fwrite( &R,sizeof( ROOM ),1,fp1 );

}

fclose( fp );

fclose( fp1 );

remove( "room.txt" );

rename( "temp.txt","room.txt" );

if( i==0 )

    printf( "Edit not completed\n" );

else

    printf( "Edit completed\n" );

printf( "Information room after Edit:\n" );

fp=fopen( "room.txt","rb" );

i=0;

while( fread( &R,sizeof( ROOM ),1,fp)==1 )

{

    printf( "%10s%20s%10s%10s\n",R.room_num,

```

```

        R.room_type,

        R.room_price,

        R.room_status );

    }

    fclose( fp );

    getch( );

}

```

### ខាងក្រោមជាលទ្ធផលទី១របស់កម្មវិធីខាងលើ

Information room before edit:

|      |             |    |      |
|------|-------------|----|------|
| R001 | normal_room | 15 | Free |
| R002 | normal_room | 15 | Free |
| R003 | vip_room    | 50 | Free |
| R004 | air_room    | 25 | Free |
| R005 | air_room    | 25 | Free |

Enter room id to edit: R000

Edit not completed

Information room after delete:

|      |             |    |      |
|------|-------------|----|------|
| R004 | air_room    | 25 | Free |
| R005 | air_room    | 25 | Free |
| R001 | normal_room | 15 | Free |
| R002 | normal_room | 15 | Free |
| R003 | vip_room    | 50 | Free |



## ខាងក្រោមជាលទ្ធផលទី២របស់កម្មវិធីខាងលើ

Information room before edit:

|      |             |    |      |
|------|-------------|----|------|
| R001 | normal_room | 15 | Free |
| R002 | normal_room | 15 | Free |
| R003 | vip_room    | 50 | Free |
| R004 | air_room    | 25 | Free |
| R005 | air_room    | 25 | Free |

Enter room id to edit: R003

Enter new room number: R003

Enter new room type: special\_room

Enter new room price: 100

Enter new room status: Free

Edit update completed

Information room after edit:

|      |              |     |      |
|------|--------------|-----|------|
| R004 | air_room     | 25  | Free |
| R005 | air_room     | 25  | Free |
| R001 | normal_room  | 15  | Free |
| R002 | normal_room  | 15  | Free |
| R003 | special_room | 100 | Free |

### XIII. លំហាត់ការគ្រប់គ្រងផ្នែកលក់ទំនិញក្នុងហាងមួយ (Update)

កម្មវិធីហាងលក់ទំនិញមានផ្នែកច្រើនក្នុងការគ្រប់គ្រងកម្មវិធីខាងក្រោមលើកយកតែផ្នែកលក់មកបង្ហាញរួមមានការងារដូចជា

- ផ្នែក Stock product
  - o Input product
  - o Show product
  - o Sort product
  - o Search product
  - o Delete product
- ផ្នែក Order product
- ផ្នែក Show order product
- ផ្នែក Delete order product
- ផ្នែក Edit order product

**សំគាល់** : ដោយប្រើ Binary File ក្នុងការផ្ទុកទិន្នន័យ

```
#include<stdio.h>
```

```
#include<conio.h>
```

```
#include<ctype.h>
```

```
#include<string.h>
```

```
#include<dos.h>
```

```
#include<stdlib.h>
```

```
struct product{
```

```
    char pid[10];
```

```
    char name[20];
```

```
    char qty[10];
```

```
    char price[10];
```

```
};
```

```
struct order{
```

```
    char rid[10];
```

```

        char pid[10];

        char qty[10];

        char price[10];

};

int i,ii,j,b,k;

char ch;

char x[20],xpid[20];

int main_menu( );

void input( );

void output( );

int search_id( char * );

void delete_id( );

void edit_id( );

int menu_pro( );

product p,p1,p2;

order  r;

FILE *fp, *fp1, *fp2, *fp3;

void main( ){

int n;

do{

        clrscr( );

        i=main_menu( );

```

```

switch(i){
case 0:

    do{

        clrscr( );

        ii=menu_pro( );

        switch(ii){

        case 0:

            do{

                clrscr( );

                printf( "Input product id: " );

                fflush( stdin );

                scanf( "%s",&p.pid );

                printf( "Input product name: " );

                scanf( "%s",&p.name );

                printf( "Input product qty : " );

                scanf( "%s",&p.qty );

                printf( "Input product price: " );

                scanf( "%s",&p.price );

                fp=fopen( "Epro.txt","ab" );

                fwrite( &p,sizeof( product ),1,fp );

                fclose( fp );

                printf( "Press key Enter to continue other back" );

```

```
}while( getch( )==13);
```

```
break;
```

case 1:

```
clrscr( );
```

```
printf( "%10s%20s%10s%10s\n","PID","Pro_name","Qty","Price" );
```

```
fp=fopen( "Epro.txt","rb" );
```

```
if( fp==NULL )
```

```
{
```

```
    printf( "File not found" );
```

```
}else{
```

```
    while( fread( &p,sizeof( product ),1,fp)==1 )
```

```
    {
```

```
        printf( "%10s%20s%10s%10s\n",p.pid,p.name,p.qty,p.price );
```

```
    }
```

```
}
```

```
getch( );
```

```
fclose( fp );
```

```
break;
```

case 2:

```
clrscr( );
```

```
fp=fopen( "Epro.txt","rb+" );
```

```
fseek( fp,0,SEEK_END );
```

```

n=ftell( fp )/sizeof( product );

rewind( fp );

for( i=0;i<n;i++ )

{

    fseek( fp,i * sizeof( product ),SEEK_SET );

    fread( &p,sizeof( product ),1,fp );

    for( j=i+1;j<n;j++ )

    {

        fseek( fp,j * sizeof( product ),SEEK_SET );

        fread( &p1,sizeof( product ),1,fp );

        if( strcmp( p.name,p1.name )>0 )

        {

            fp1=fopen( "Epro.txt","rb" );

            fp2=fopen( "temp1.txt","wb" );

            for( k=0;k<n;k++ )

            {

                fread( &p2,sizeof( product ),1,fp1 );

                if( k==i )

                {

                    p2 = p1;

                }

                else if( k==j )

```

```

        {

            p2 = p;

        }

        fwrite( &p2,sizeof( product ),1,fp2 );

    }

    fclose( fp1 );

    fclose( fp2 );

    remove( "Epro.txt" );

    rename( "temp1.txt","Epro.txt" );

    fp=fopen( "Epro.txt","rb+" );

    p=p1;

    }

}

gotoxy( 1,13 );

printf( "Sort completed" );

getch( );

fclose( fp );

    break;

case 3:

    clrscr( );

    printf( "Please Enter product name to search: " );

```

```

scanf( "%s",&x);

printf( "%10s%20s%10s%10s\n","PID","Pro_name","Qty","Price" );

fp=fopen( "Epro.txt","rb" );

if( fp==NULL )

{

    printf( "File not found" );

}else

{

    b=0;

    while( fread( &p,sizeof( product ),1,fp)==1 )

    {

        if( strcmp( p.name,x)==0 )

        {

            b=1;

printf( "%10s%20s%10s%10s\n",p.pid,p.name,p.qty,p.price);

        }

    }

}

if( b==0 )

    printf( "product name not found" );

getch( );

fclose( fp );

break;

```



case 4:

```
clrscr( );

printf( "Enter product id to delete: " );

fflush( stdin );

gets( x );

b=0;

fp=fopen( "Epro.txt","r" );

fp1=fopen( "temp.txt","w" );

while( fread( &p,sizeof( product ),1,fp )==1 )

{

    if( strcmp( p.pid,x )!=0 )

    {

        fwrite( &p,sizeof( product ),1,fp1 );

    }else

        b=1;

}

fclose( fp1 );

fclose( fp );

remove( "Epro.txt" );

rename( "temp.txt","Epro.txt" );

if( b==0 )

    printf( "Delete product not completed " );
```

```

        else

            printf( "Delete completed" );

            getch( );

            break;

    }

}while( ii !=5 );

break;

case 1:

    input( );

    break;

case 2:

    output( );

    break;

case 3:

    textbackground( BLUE );

    clrscr( );

    printf( "Enter order id to Search : " );

    scanf( "%s",&x );

    b=search_id( x );

    if( b==0 ){

        printf( "Order id not found" );

    }

```

```

        getch( );

        break;

case 4:

        delete_id( );

        break;

case 5:

        edit_id( );

        break;

    }

}while(i!=6);

}

void input( ){

float k;

        textbackground( BLUE );

do{

again:

        clrscr( );

        gotoxy( 1,25 );

        printf( "Enter order id='exit' to stop " );

        gotoxy( 1,1 );

        printf( "Enter order id: " );

        scanf( "%s",&r.riid );

```

```

if( strcmp( r.rid,"exit")==0)

{

    break;

}

printf( "Enter product id: ");

scanf( "%s",&r.pid );

b=0;

fp=fopen( "Epro.txt","rb" );

fp1=fopen( "temp.txt","wb" );

while( fread( &p,sizeof( product ),1,fp)==1 )

{

    if( strcmp( p.pid,r.pid)==0){

        b=1;

        printf( "Enter order qty: ");

        scanf( "%s",&r.qty );

        printf( "Enter order price: ");

        scanf( "%s",&r.price );

        itoa( atoi( p.qty )-atoi( r.qty ),p.qty,10 );

        fp2=fopen( "Eorder.txt","a" );

        fwrite( &r,sizeof( order ),1,fp2 );

        fclose( fp2 );

    }

}

```

```

fwrite( &p,sizeof( product ),1,fp1 );

}

fclose( fp );

fclose( fp1 );

remove( "Epro.txt" );

rename( "temp.txt","Epro.txt" );

if( b==0 )

{

    gotoxy( 1,13 );

    printf( "Product id not found try again!" );

    getch( ); goto again;

}

printf( "Press Enter continue other exit" );

}while( getch( )==13 );

}

void output( ){

textbackground( BLUE );

clrscr( );

printf( "%10s%20s%10s%10s\n","RID","Pro_id","Qty","Price" );

fp=fopen( "Eorder.txt","rb" );

if( fp==NULL )

{

```

```

        printf( "File not found" );

    }else

    {

        while( fread( &r,sizeof( order ),1,fp)==1 )

        {

            printf( "%10s%20s%10s%10s\n",r.rid,r.pid,r.qty,r.price );

        }

    }

    getch( );

    fclose( fp );

}

int search_id( char *x){

    b=0;

    fp=fopen( "Eorder.txt","rb" );

    if( fp==NULL )

    {

        printf( "File not found" );

    }

    else

    {

        while( fread( &r,sizeof( order ),1,fp)==1 )

        {

```

```

        if( strcmp( r.rid,x)==0 ){

            if( b==0 )

                printf( "%10s%20s%10s%10s\n","RID","Pro_id","Qty","Price" );

            b=1;

            printf( "%10s%20s%10s%10s\n",r.rid,r.pid,r.qty,r.price );

        }

    }

    fclose( fp );

    return b;

}

void delete_id( ){

    textbackground( BLUE );

    clrscr( );

    printf( "Enter order id to delete: " );

    scanf( "%s",&x );

    printf( "Enter product id: " );

    scanf( "%s",&xp );

    b=0;

    fp=fopen( "Eorder.txt","rb" );

    fp1=fopen( "temp.txt","wb" );

    while( fread( &r,sizeof( order ),1,fp )==1 )

```

```

{

    if( strcmp( r.rid,x)==0 && strcmp( r.pid,xpid)==0 )

    {

        b=1;

        fp2=fopen( "Epro.txt","rb" );

        fp3=fopen( "temp1.txt","wb" );

        while( fread( &p,sizeof( product ),1,fp2)==1 )

        {

            if( strcmp( p.pid,r.pid)==0 )

            {

                itoa( atoi( p.qty )+atoi( r.qty ),p.qty,10 );

            }

            fwrite( &p,sizeof( product ),1,fp3 );

        }

        fclose( fp2 );

        fclose( fp3 );

        remove( "Epro.txt" );

        rename( "temp1.txt","Epro.txt" );

    }else

    {

        fwrite( &r,sizeof( order ),1,fp1 );

    }
}

```



```

    }

    fclose( fp );

    fclose( fp1 );

    remove( "Eorder.txt" );

    rename( "temp.txt", "Eorder.txt" );

    if( b==0 )

        printf( "Delete not completed" );

    else

        printf( "Delete completed" );

    getch( );
}

void edit_id( ){

    textbackground( BLUE );

    clrscr( );

    printf( "Enter Order id to Edit: " );

    scanf( "%s",&x );

    printf( "Enter product id to Edit: " );

    scanf( "%s",&xp );

    b=0;

    fp=fopen( "Eorder.txt", "rb" );

    fp1=fopen( "temp.txt", "wb" );

    while( fread( &r, sizeof( order ), 1, fp ) == 1 )

```

```

{

    if( strcmp( r.rid,x)==0 && strcmp( r.pid,xpid)==0 )

    {

        b=1;

        fp2=fopen( "Epro.txt","rb" );

        fp3=fopen( "temp1.txt","wb" );

        while( fread( &p,sizeof( product ),1,fp2)==1 )

        {

            if( strcmp( p.pid,r.pid)==0 )

            {

                itoa( atoi( p.qty )+atoi( r.qty ),p.qty,10 );

            }

            fwrite( &p,sizeof( product ),1,fp3 );

        }

        fclose( fp2 );

        fclose( fp3 );

        remove( "Epro.txt" );

        rename( "temp1.txt","Epro.txt" );

        printf( "Enter new product id " );

        scanf( "%s",&r.pid );

        printf( "Enter new order qty: " );

        scanf( "%s",&r.qty );

```

```

printf( "Enter new price: " );

scanf( "%s",&r.price );

fp2=fopen( "Epro.txt","rb" );

fp3=fopen( "temp1.txt","wb" );

while( fread( &p,sizeof( product ),1,fp2 )==1 )

{

    if( strcmp( p.pid,r.pid )==0 )

    {

        itoa( atoi( p.qty )-atoi( r.qty ),p.qty,10);

    }

    fwrite( &p,sizeof( product ),1,fp3 );

}

fclose( fp2 );

fclose( fp3 );

remove( "Epro.txt" );

rename( "temp1.txt","Epro.txt" );

}

fwrite( &r,sizeof( order ),1,fp1 );

}

fclose( fp );

fclose( fp1 );

remove( "Eorder.txt" );

```

```

rename( "temp.txt", "Eorder.txt" );

if( b==0)

    printf( "Edit not completed" );

else

    printf( "Edit Update completed" );

getch( );

}

int main_menu( ){

do{

    textbackground( BLUE );

    clrscr( );

    textcolor( WHITE );

    textbackground( GREEN );

    gotoxy( 1,9+i);

    cprintf( "%30c", ' ');

    gotoxy( 1,9);

    printf( "Check product in Stock" );

    gotoxy( 1,10 );

    printf( "Register information student" );

    gotoxy( 1,11 );

    printf( "Show information student" );

    gotoxy( 1,12 );

```

```

printf( "Search information student" );

gotoxy( 1,13 );

printf( "Delete information student" );

gotoxy( 1,14 );

printf( "Edit information student" );

gotoxy( 1,15 );

printf( "Exit program\n" );

gotoxy( 1,25 );

printf( "Use key to Scroll 'A' move 'UP' or 'Z' move 'Down'" );

printf( " Enter='Select'" );

ch=getch( );

switch( tolower( ch ) ){

case 'a':

    i=i-1;

    if( i<0 ){

        i=6;

    }

    break;

case 'z':

    i=i+1;

    if( i>6 )

    {

```

```

                i=0;

            }

        }

}while( ch!=27 && ch!=13);

if( ch==27)

{

    i=6;

}

return i;

}

int menu_pro( ){

do{

    textbackground( BLUE );

    clrscr( );

    textcolor( WHITE );

    textbackground( GREEN );

    gotoxy( 1,10+i);

    cprintf( "%30c",' ');

    gotoxy( 1,10 );

    printf( "Input product in stock" );

    gotoxy( 1,11 );

    printf( "Show product in stock" );

```

```

gotoxy( 1,12 );

printf( "Sort product in stock( name )" );

gotoxy( 1,13 );

printf( "Search product in stock" );

gotoxy( 1,14 );

printf( "Delete product in stock" );

gotoxy( 1,15 );

printf( "Exit program\n" );

gotoxy( 1,25 );

printf( "Use key to Scroll 'A' move 'UP' or 'Z' move 'Down'" );

printf( " Enter='Select'" );

ch=getch( );

switch( tolower( ch ) )

{

case 'a':

    i=i-1;

    if( i<0 ) i=5;

    break;

case 'z':

    i=i+1;

    if( i>5 ) i=0;

}

```

```
}while( ch!=27 && ch!=13);  
  
if( ch==27) i=5;  
  
return i;  
  
}
```



## **ឯកសារពិគ្រោះ**

<https://www-personal.acfr.usyd.edu.au/tbailey/ctext/ctext.pdf>

[https://www.unf.edu/~wkloster/2220/ppts/cprogramming\\_tutorial.pdf](https://www.unf.edu/~wkloster/2220/ppts/cprogramming_tutorial.pdf)

[https://cimec.org.ar/~mstorti/C/CURSOS/ccourse\\_fix.pdf](https://cimec.org.ar/~mstorti/C/CURSOS/ccourse_fix.pdf)

<https://www.tutorialspoint.com/cprogramming/index.htm>