សាកលវិទ្យាល័យជាតិគ្រប់គ្រង

National University of Management

គណិតវិទ្យាកុំព្យូទ័រ

ឆាយ ផាង   លី បូ៉ាឌុន   ចាប ឌីណា

២០២១

សាកលវិទ្យាល័យជាតិគ្រប់គ្រង

National University of Management

គណិតវិទ្យាកុំព្យូទ័រ

ឆាយ ផាង   លី បូរ៉ាឌុន   ចាប ឌីណា

២០២១

បុព្វកថា

ដំណើរអភិវឌ្ឍនៃព្រះរាជាណាចក្រកម្ពុជានៅក្នុងយុគសម័យទំនើបនេះ ជាមេរៀនដ៏ជោគជ័យបំផុតមួយ ដែលចាប់ឫសគល់ចេញពីការបញ្ចប់របបប្រល័យពូជសាសន៍ ការបញ្ចប់សង្គ្រាម ការផ្សះផ្សារជាតិ ការកសាងមូលដ្ឋានរឹងមាំនៃសន្តិភាពនិងស្ថេរភាព និងការអភិវឌ្ឍសេដ្ឋកិច្ច។ នៅក្រោយពេលដែលសន្តិភាពត្រូវបានកើតឡើងដោយបរិបូរណ៍នៅឆ្នាំ១៩៩៨ កម្ពុជាទទួលកំណើនសេដ្ឋកិច្ចខ្ពស់ គឺប្រមាណ៨% ក្នុងមួយឆ្នាំ។ លើសពីនេះទៀត អត្រានៃភាពក្រីក្រត្រូវបានកាត់បន្ថយពីប្រមាណ៥៣% នៅឆ្នាំ២០០៤ មកនៅទាបជាង១០% នៅឆ្នាំ២០១៩។ ដំណើរនៃការអភិវឌ្ឍជាតិជាសកម្មភាពដែលបន្តទៅមុខជាប់ជានិច្ច ហើយគោលនយោបាយថ្មីៗដែលមានលក្ខណៈអន្តរវិស័យគ្របដណ្តប់ក៏កំពុងលេចរូបរាងឡើង ដើម្បីតម្រង់ទិសកម្ពុជាឆ្ពោះទៅកាន់ប្រទេសមានប្រាក់ចំណូលមធ្យមកម្រិតខ្ពស់នៅឆ្នាំ២០៣០ និងឈានឡើងជាប្រទេសមានប្រាក់ចំណូលខ្ពស់ នៅឆ្នាំ២០៥០។ ការប្រែប្រួលឆាប់រហ័សនៃនិម្មាបនកម្មពិភពលោកនិងតំបន់ រួមទាំងទំនាក់ទំនងភូមិសាស្ត្រនយោបាយ បានផ្តល់កាលានុវត្តភាពសម្រាប់ការអភិវឌ្ឍឧស្សាហកម្មនៅកម្ពុជា ដែលត្រូវបានរាជរដ្ឋាភិបាលចាត់ទុកជាមូលដ្ឋានគ្រឹះនៃកំណើនសេដ្ឋកិច្ចកម្ពុជា។ រាជរដ្ឋាភិបាលកម្ពុជាបាន និងកំពុងបន្តពង្រឹងនិងអភិវឌ្ឍវិស័យអប់រំឆ្ពោះទៅរកការស្រាវជ្រាវនិងនវានុវត្តន៍ ដើម្បីពង្រឹងសមត្ថភាពនិងជំនាញរបស់ធនធានមនុស្សនៅកម្ពុជា ឱ្យស្របទៅនឹងបរិបទថ្មីនៃការអភិវឌ្ឍ ជាពិសេសការពង្រឹងសហគ្រិនភាពក្នុងការរៀបចំម៉ូដែលធុរកិច្ចថ្មីៗ។ ដើម្បីចាប់យកកាលានុវត្តភាពពីបដិវត្តន៍ឧស្សាហកម្មទី៤ និងសេដ្ឋកិច្ចឌីជីថលដែលកំពុងផុសផុលឡើង ប្រព័ន្ធអេកូឡូហ្ស៊ីដែលបង្កលក្ខណៈអំណោយផលដល់ការបង្កើតថ្មី នវានុវត្តន៍ ការស្រាវជ្រាវ និងអភិវឌ្ឍន៍ត្រូវតែមានការកែលម្អ។

បណ្តាប្រទេសនៅទ្វីបអាស៊ីកំពុងនាំមុខក្នុងការវិនិយោគលើការស្រាវជ្រាវនិងអភិវឌ្ឍ ដោយមានភាគហ៊ុនប្រមាណ៤៤% នៃការវិនិយោគទាំងមូលរបស់ពិភពលោក។ ប្រទេសចិនកំពុងបន្តកសាងហេដ្ឋារចនាសម្ព័ន្ធនៃការវិនិយោគលើការស្រាវជ្រាវនិងអភិវឌ្ឍ ក៏ដូចជាសមត្ថភាពមនុស្ស។ ផ្ទុយទៅវិញ ប្រទេសនៅទ្វីបអាមេរិកខាងត្បូងនិងអាហ្វ្រិក កំពុងស្ថិតនៅឆ្ងាយពីការវិនិយោគនេះ ហើយជាលទ្ធផលប្រទេសទាំងនោះក៏ពុំមានកំណើនសេដ្ឋកិច្ចគួរឱ្យកត់សម្គាល់ដែរ។ ទុនវិនិយោគសរុបលើការស្រាវជ្រាវនិងអភិវឌ្ឍរបស់ប្រទេសនៅទ្វីបអាមេរិកខាងត្បូងនិងអាហ្វ្រិក មានប្រមាណ៥%នៃការវិនិយោគទាំងមូលរបស់ពិភពលោក ក្នុងពេលដែលតំបន់ទាំង២នេះមានប្រជាជនប្រមាណ២០%នៃប្រជាជនពិភពលោក។ ប្រទេសចំនួន៦ដែលមានលំដាប់ខ្ពស់ជាងគេនៅក្នុងការវិនិយោគលើការស្រាវជ្រាវនិងអភិវឌ្ឍ រួមមានសហរដ្ឋអាមេរិក ចិន ជប៉ុន អាល្លឺម៉ង់ ឥណ្ឌា និងកូរ៉េខាងត្បូង ដែលស្មើនឹងប្រមាណ៧០%នៃទុនវិនិយោគសរុបរបស់ពិភពលោក។

តើចំណេះដឹង ផលិតផល និងសេវាកម្មថ្មីទាំងនេះកើតឡើងពីអ្វី? ហើយកើតឡើងដោយរបៀបណា? ព្រះរាជាណាចក្រកម្ពុជាកំពុងតែកសាងមូលដ្ឋានសម្រាប់ការត្រៀមខ្លួនទទួល និងប្រកួតប្រជែងក្នុងយុគសម័យបដិវត្តឧស្សាហកម្មទី៤ នៅក្នុងសេដ្ឋកិច្ចដែលផ្អែកលើពុទ្ធិ ហើយដែលប្រការនេះ

ចាំបាច់តម្រូវឱ្យពលរដ្ឋកម្ពុជា ត្រូវក្លាយខ្លួនជាពលរដ្ឋឌីជីថល ពលរដ្ឋសកល និងពលរដ្ឋដែលប្រកបដោយការទទួលខុសត្រូវ ដែលមានសមត្ថភាពក្នុងការផលិត ចែកចាយ និងប្រើប្រាស់ពុទ្ធិដើម្បីទទួលមនុញ្ញផល និងរួមចំណែកក្នុងកំណើន។ ធនាគារពិភពលោកបានធ្វើការកត់សម្គាល់តាំងពីឆ្នាំ២០០២នូវបម្លាស់ប្តូរនៃមូលដ្ឋានសេដ្ឋកិច្ច ពីសេដ្ឋកិច្ចដែលពឹងផ្អែកលើកម្លាំងពលកម្ម និងធនធានអតិកម្ម (Labour and Resource Based Economy) ទៅកាន់សេដ្ឋកិច្ចដែលពឹងផ្អែកលើពុទ្ធិ (Knowledge Based-Economy) ដែលក្នុងន័យនេះ ពុទ្ធិគឺជាគន្លឹះនៃការអភិវឌ្ឍ។ អាស្រ័យហេតុនេះ នៅលើគន្លងដែលកម្ពុជាកំពុងធ្វើដំណើរឆ្ពោះទៅកាន់សេដ្ឋកិច្ចឌីជីថល សង្គមកម្ពុជាត្រូវតែមានសមត្ថភាពក្នុងការផលិត ជ្រើសរើស បន្សាំ បង្កើតមុខរបរ និងប្រើប្រាស់ពុទ្ធិ ដើម្បីរក្សានិរន្តរភាពនៃកំណើន និងកែលម្អជីវភាពរស់នៅ។ សមត្ថភាពទាំងនេះ អាចកើតឡើងនៅពេលពលរដ្ឋកម្ពុជាមានឱកាសក្នុងការទទួលបានបទពិសោធន៍ពីការស្រាវជ្រាវ ការបណ្តុះគំនិតច្នៃប្រឌិត និងការស្វែងរកនវានុវត្តន៍។

កំណែទម្រង់វិស័យអប់រំ គឺជាការត្រួសត្រាយមាគ៌ាសម្រាប់ដំណើរឆ្ពោះទៅកាន់សង្គមប្រកបដោយពុទ្ធិ និងប្រជាពលរដ្ឋប្រកបដោយភាពរស់រវើក។ តាមរយៈមូលដ្ឋានអប់រំ សង្គមប្រកបដោយពុទ្ធិនឹងប្រមូលផ្តុំ បង្កើត និងចែករំលែក ទៅកាន់សមាជិកក្នុងសង្គមនូវសម្បទាអប់រំ ពិសេសគឺពុទ្ធិសម្បទាក្នុងបុព្វហេតុនៃមនុស្សជាតិនិងឧត្តមប្រយោជន៍នៃប្រទេស។ សង្គមប្រកបដោយពុទ្ធិ គឺពុំគ្រាន់តែជាសង្គមដែលសម្បូរព័ត៌មានប៉ុណ្ណោះទេ តែជាសង្គមដែលប្រជាពលរដ្ឋអាចធ្វើបរិវត្តកម្មពីព័ត៌មានទៅជាមូលធនប្រកបដោយប្រសិទ្ធភាព។ ការរីកចម្រើនទៅមុខជាលំដាប់នៃបច្ចេកវិទ្យានិងតំណភ្ជាប់ បានពង្រីកព្រំដែននៃការចូលទៅកាន់ និងការទទួលបានព័ត៌មានជាសកល ហើយដែលក្នុងន័យនេះ ការអប់រំនឹងបន្តវិវត្តទៅមុខនិងមានការផ្លាស់ប្តូរ។ សង្គមមួយដែលមានអំណាន និងរបាប់ជាបុរេលក្ខខណ្ឌនៃជីវភាពប្រចាំថ្ងៃនៃប្រជាពលរដ្ឋ ពេលនោះបំណិននៃអំណាន និពន្ធ និងការគណនាលេខនព្វន្ត គឺជាចលករនៃការរៀនរបស់សិស្ស។ ជាតុដ៏ចម្បងមួយដែលស្ថិតនៅក្នុងការកសាងសង្គមដែលប្រកបដោយពុទ្ធិគឺសៀវភៅសិក្សា ហើយការរៀបរៀង និពន្ធ និងកែលម្អសៀវភៅសិក្សាជាប្រចាំ គឺជានវានុវត្តន៍នៃវិស័យអប់រំដែលនាំទៅរកការសិក្សាពេញមួយជីវិត ការអភិវឌ្ឍសម្បទាអប់រំ និងការចែករំលែកចំណេះដឹង។ មូលដ្ឋានអប់រំ ជាពិសេសគឺគ្រឹះស្ថានឧត្តមសិក្សាត្រូវមានតួនាទីដែលប្រកបដោយការឆ្លើយតប ចំពោះតម្រូវការខាងលើនេះ។ សាស្ត្រាចារ្យ អ្នកស្រាវជ្រាវ និងបុគ្គលិកអប់រំត្រូវបន្តសិក្សាជាប់ជានិច្ច តាមរយៈការរៀបរៀង និពន្ធ និងកែលម្អសៀវភៅសិក្សា ហើយដែលសៀវភៅសិក្សាទាំងនេះនឹងក្លាយជាស្ពាននៃទំនាក់ទំនងរវាងនវានុវត្តន៍នៃបច្ចេកវិទ្យា និងការរៀននិងបង្រៀននៅក្នុងថ្នាក់រៀន។

សង្គមដែលប្រកបពុទ្ធិ ក៏ជាសង្គមដែលបណ្តុះឱ្យមានរចនាសម្ព័ន្ធទន់នៃសេដ្ឋកិច្ចដែលពឹងផ្អែកលើពុទ្ធិដែរ។ ឧទាហរណ៍ជាក់ស្តែងនៃបែបផែននេះរួមមាន Silicon Valley នៃសហរដ្ឋអាមេរិកសួនឧស្សាហកម្មវិស្វកម្មអាកាសយានយន្តនិងយានយន្តនៅទីក្រុង Munich ប្រទេសអាល្លឺម៉ង់ តំបន់ជីវបច្ចេកវិទ្យានៅក្រុង Hyderabad ប្រទេសឥណ្ឌា តំបន់ផលិតគ្រឿងអេឡិចត្រូនិកនិងសារគមនាគមន៍ឌីជីថលនៅទីក្រុង Seoul ប្រទេសកូរេខាងត្បូង ក៏ដូចជាសួនឧស្សាហកម្មថាមពល និងឥន្ធនគីមីសាស្ត្រនៃប្រទេសប្រេស៊ីល ហើយក៏នៅមានទីក្រុងនៃប្រទេសជាច្រើនទៀតនៅលើពិភពលោក។ លក្ខណៈសម្បត្តិ

នៃទីក្រុងទាំងនេះគឺការប្រើប្រាស់និន្នាការនៃការអភិវឌ្ឍដែលជំរុញ និងតម្រង់ទិសដោយចំណេះដឹង ហើយដែលចំណេះដឹងទាំងនោះកើតចេញជាដំបូងពីការវិនិយោគទៅលើគ្រឹះស្ថានឧត្តមសិក្សា ស្ថាប័នស្រាវជ្រាវ មជ្ឈមណ្ឌលឧត្តមភាពនៃជំនាញជាន់ខ្ពស់ ការប្រកួតប្រជែងដោយគុណាធិបតេយ្យ និងជាពិសេសគឺការបណ្ដុះវប្បធម៌អំណាននិងនិពន្ធសៀវភៅ។ ល្បឿននៃការរីកចម្រើនផ្នែកពុទ្ធិ និងបច្ចេកវិទ្យាកំពុងមានសន្ទុះលឿនជាងអ្វីដែលសិស្ស និងនិស្សិតអាចទទួលបានពីគ្រូនៅគ្រឹះស្ថានសិក្សា ដែលធ្វើឱ្យគោលដៅនៃការអប់រំនៅពេលបច្ចុប្បន្ននេះ មានការប្រឈមខ្លាំងជាងពេលណាទាំងអស់។ ឧទាហរណ៍ក្នុងមួយឆ្នាំ មានសៀវភៅជាង២,២លានចំណងជើង ត្រូវបានសរសេរនិងបោះពុម្ព ដែលក្នុងនោះប្រទេសចិនមាន៤៤០ពាន់ ចំណែកឯសហរដ្ឋអាមេរិកមាន៣០៥ពាន់ និងប្រទេសរុស្ស៊ីមាន១២០ពាន់ចំណងជើង។

ខណៈពេលដែលបច្ចេកវិទ្យាកំពុងរីកចម្រើនជារៀងរាល់ថ្ងៃ មធ្យោបាយសម្រាប់អំណានក៏មានច្រើនជម្រើសសម្រាប់សិស្ស-និស្សិត និងសាធារណៈជន រួមមានការអានសៀវភៅ ការអានលើឧបករណ៍អេឡិចត្រូនិក ការអានដោយប្រើទូរសព្ទវៃឆ្លាត និងការអានលើកុំព្យូទ័រ ដែលសុទ្ធសឹងជាមធ្យោបាយសំខាន់ៗដែលនាំអ្នកអានទាំងឡាយឱ្យសម្រេចគោលបំណងអានរបស់ខ្លួន។ ម្យ៉ាងវិញទៀត អំណានដោយប្រើមធ្យោបាយបច្ចេកវិទ្យាទំនើប ចំណាយពេលតិច ងាយស្រួលអាន និងជួយដល់បរិស្ថានមួយកម្រិតទៀត។ នាពេលបច្ចុប្បន្ន សិស្ស-និស្សិត និងសាធារណៈជនកម្ពុជាដែលស្រឡាញ់អំណានកំពុងតែប្រើប្រាស់មធ្យោបាយអំណានទាំងនេះ។ បើយើងក្រឡេកមើលទៅប្រទេសជឿនលឿន ទោះបីជាបច្ចេកវិទ្យារីកចម្រើនខ្លាំងយ៉ាងណា អំណានតាមរយៈសៀវភៅនៅតែមានសន្ទុះដដែល។ ម្យ៉ាងវិញទៀត បច្ចេកវិទ្យាអានបែបទំនើបតាមរយៈឧបករណ៍ទំនើប អាស្រ័យលើលទ្ធភាពនៃធនធានអប់រំឌីជីថល និងមាតិកាឌីជីថលគ្រប់គ្រាន់ដែលបានផលិត និងបង្ហោះចែកចាយសម្រាប់អំណាន។

ក្នុងបរិបទកម្ពុជា ជាពិសេសក្នុងបរិការណ៍នៃការផ្ទុះរីករាលដាលនៃជំងឺកូវីដ-១៩ ក្រសួងអប់រំ យុវជន និងកីឡា បានជំរុញឱ្យមានបរិវត្តកម្មឌីជីថលនៅក្នុងអេកូស៊ីស្ដែមនៃការអប់រំ ជាពិសេសការអប់រំតាមប្រព័ន្ធអេឡិចត្រូនិកនិងការអប់រំពីចម្ងាយ ដើម្បីលើកកម្ពស់អំណាន តាមរយៈការផលិតមាតិកាឌីជីថលដែលមានភាពចំរុះ ការកសាងសមត្ថភាពផ្នែកតំណភ្ជាប់និងវេទិកាឌីជីថល ការពង្រីកវិសាលភាពនៃមជ្ឈមណ្ឌលទិន្នន័យ និងការលើកកម្ពស់គុណភាពនៃការផលិតធនធានអប់រំឌីជីថល គួបផ្សំជាមួយការចែកសន្លឹកកិច្ចការឱ្យសិស្សយកទៅរៀននៅផ្ទះ និងការចុះទៅជួបជាមួយសិស្សជាបណ្ដុំនៅតាមសហគមន៍។ ក្នុងន័យលើកកម្ពស់អំណាន និងភាពសម្បូរបែបនៃធនធានសៀវភៅសិក្សា ឱ្យកាន់តែមានប្រសិទ្ធភាពនិងភាពសក្តិសិទ្ធិ និងផ្ដល់ឱកាសអំណានកាន់តែច្រើនថែមទៀតដល់សិស្សានុសិស្ស និស្សិត និងសាធារណៈជន ក្រសួងអប់រំ យុវជន និងកីឡាលើកទឹកចិត្តនូវចំណុចមួយចំនួនដូចខាងក្រោម៖

១. សាស្ត្រាចារ្យ អ្នកស្រាវជ្រាវ និងបុគ្គលិកអប់រំ សូមបន្តនិងបង្កើនការបោះពុម្ពស្នាដៃបន្ថែមទៀត ដើម្បីធ្វើឱ្យធនធានសម្រាប់អំណានកាន់តែសម្បូរបែប ជាពិសេសធនធានអំណានជាខេមរភាសា

២. គ្រឹះស្ថានឧត្តមសិក្សា សូមផ្តល់លទ្ធភាពគ្រប់បែបយ៉ាង ដើម្បីឱ្យបុគ្គលិកអប់រំគ្រប់លំដាប់ថ្នាក់ និងនិស្សិតគ្រប់កម្រិតសិក្សាអាចចូលរួមអាន និងសិក្សាស្រាវជ្រាវតាមគ្រប់លទ្ធភាពជាមួយធនធានអំណាន ជាពិសេសការរៀបចំឱ្យមានពេលវេលាសម្រាប់សហាសិក្សា និងអំណានក្នុងបណ្ណាល័យ
៣. សាស្ត្រាចារ្យតាមមុខវិជ្ជា និងអ្នកស្រាវជ្រាវតាមជំនាញឬវិស័យ ត្រូវរៀបចំដំណើរការរៀនបង្រៀន និងស្រាវជ្រាវដែលមានដាក់បញ្ចូលកិច្ចការស្វ័យសិក្សា សហាសិក្សា ឬការស្រាវជ្រាវបណ្ណាល័យដែលតម្រូវឱ្យនិស្សិត ត្រូវអាននិងស្រាវជ្រាវជាមួយធនធានអំណាន
៤. គ្រឹះស្ថានឧត្តមសិក្សា និងមជ្ឈមណ្ឌលស្រាវជ្រាវ ត្រូវខិតខំឱ្យអស់លទ្ធភាពក្នុងការបង្កើតបណ្ណាល័យ មជ្ឈមណ្ឌលរក្សាឯកសារ ឬមជ្ឈមណ្ឌលអប់រំឌីជីថលជាដើម ដើម្បីឱ្យបុគ្គលិកអប់រំគ្រប់លំដាប់ថ្នាក់និងនិស្សិតគ្រប់កម្រិតសិក្សាអាចទទួលបាន និងស្វែងរកប្រភពសម្រាប់អំណានកាន់តែសម្បូរបែប និងមានភាពបត់បែន ឆ្លើយតបតាមតម្រូវការអ្នកអាន
៥. និស្សិតគ្រប់កម្រិតសិក្សាត្រូវខិតខំនិងចំណាយពេលវេលាដើម្បីអាន និងចាត់ទុកវប្បធម៌និងអកប្បកិរិយាអំណានជាផ្នែកមួយ នៃពេលវេលានិងភាពស៊ីវិល័យនៃជីវិតប្រចាំថ្ងៃ
៦. បងប្អូនជនរួមជាតិ ដែលជាមាតាបិតា ឬអ្នកអាណាព្យាបាល សូមជួយជំរុញនិងបង្កលក្ខណៈកាន់តែច្រើនថែមទៀត ជាពិសេសការលៃលកចំណាយនៅក្នុងគ្រួសារសម្រាប់ការទិញសម្ភារៈសិក្សា សៀវភៅអាន និងឧបករណ៍សម្រាប់អំណានដល់កូនៗ ដែលចាត់ទុកជាការវិនិយោគមួយដ៏សំខាន់ សម្រាប់ បង្កើនចំណេះដឹង និងអនាគតរបស់ពួកគេ។

ដោយមានការគាំទ្រពីក្រសួងសេដ្ឋកិច្ច និងហិរញ្ញវត្ថុ នៅឆ្នាំ២០២០ ក្រសួងអប់រំ យុវជន និងកីឡា បានបង្កើតមូលនិធិស្រាវជ្រាវ គំនិតច្នៃប្រឌិត និងនវានុវត្តន៍ ដែលហៅកាត់ថា “មូលនិធិ ស.គ.ន.” និងហៅជាភាសាអង់គ្លេសថា The Research Creativity and Innovation Fund ដែលហៅកាត់ជាភាសាអង់គ្លេសថា “RCI Fund”។ គោលដៅចម្បងនៃមូលនិធិនេះ គឺរួមចំណែកលើកកម្ពស់វប្បធម៌នៃការស្រាវជ្រាវ បំផុសគំនិតច្នៃប្រឌិត និងជំរុញការធ្វើនវានុវត្ត ដើម្បីជាប្រយោជន៍ដល់វិស័យអប់រំ យុវជន និងកីឡា ដែលឆ្លើយតបទៅនឹងទីផ្សារពលកម្ម និងសាកលភាវូបនីយកម្ម។ មូលនិធិ ស.គ.ន. បានសម្រេចកំណត់ប្រធានបទ ជាអាទិភាពសម្រាប់ការគាំទ្រដោយមូលនិធិចំនួន៣ រួមមានឌីជីថលនីយកម្មសម្រាប់បដិវត្តឧស្សាហកម្ម៤.០ (Digitalization for IR.4.0) ការស្រាវជ្រាវអនុវត្តលើវិស័យកសិកម្ម (Applied Agricultural Research) និងការស្រាវជ្រាវគរុកោសល្យសតវត្សទី២១ (21st Century Pedagogy Research)។

ដោយមានការធ្វើអាទិភាពរូបនីយកម្មទៅលើទិសដៅ នៃការប្រើប្រាស់ថវិកាមូលនិធិសម្រាប់ឆ្នាំ២០២០ ក្រសួងសេដ្ឋកិច្ច និងហិរញ្ញវត្ថុ និងក្រសួងអប់រំ យុវជន និងកីឡា បានផ្តល់ការគាំទ្រដល់ការ**រៀបរៀង និពន្ធ និងកែលម្អ សៀវភៅសិក្សា (Text book) ដែលនឹងត្រូវប្រើប្រាស់នៅកម្រិតឧត្តមសិក្សា។** គោលបំណងនៃការរៀបរៀង និពន្ធ និងកែលម្អ សៀវភៅសិក្សានៅកម្រិតឧត្តមសិក្សា គឺដើម្បីបង្កើនបរិមាណ លើកកម្ពស់គុណភាព និងពង្រីកសមធម៌នៃធនធានសិក្សាជាខេមរភាសា ជូនដល់និស្សិត

ដែលកំពុងបន្តការសិក្សា និងត្រៀមខ្លួនធ្វើការស្រាវជ្រាវនៅកម្រិតឧត្តមសិក្សា។ លើសពីនេះទៀត ការរៀបរៀង និពន្ធ និងកែលម្អសៀវភៅសិក្សានៅកម្រិតឧត្តមសិក្សា មានគោលដៅដូចខាងក្រោម ៖

១. ឆ្លើយតបជាបន្ទាន់ចំពោះការខ្វះខាតធនធានសិក្សា ដែលជាតម្រូវការសិក្សារបស់និស្សិតនៅកម្រិតឧត្តមសិក្សា
២. លើកកម្ពស់ទំនើបភាវូបនីយកម្ម និងឧត្តមានុវត្តន៍នៃការរៀននិងបង្រៀន និងការស្រាវជ្រាវនៅលើមុខវិជ្ជា កម្មវិធីសិក្សា ឬមុខជំនាញជាក់លាក់
៣. បង្កើនភាពស៊ីជម្រៅក្នុងការកសាងវិជ្ជាជីវៈនិងបទពិសោធន៍សម្រាប់បានសាស្ត្រាចារ្យ និងអ្នកស្រាវជ្រាវ
៤. រួមចំណែកដល់ការកសាងភាពជាសហគមន៍វិជ្ជាជីវៈ ការចែករំលែកបទពិសោធន៍ និងវប្បធម៌នៃការរៀបរៀង និពន្ធ និងកែលម្អសៀវភៅសិក្សានៅកម្រិតឧត្តមសិក្សា។

ក្រសួងអប់រំ យុវជន និងកីឡា បានវាយតម្លៃខ្ពស់ចំពោះការបោះជំហានប្រកបដោយមនសិការវិជ្ជាជីវៈនៃគ្រឹះស្ថានឧត្តមសិក្សា និងបុគ្គលិកអប់រំទាំងអស់ ក្នុងការរៀបចំ រៀបរៀង និពន្ធ និងកែលម្អសៀវភៅសិក្សា ដើម្បីបង្កើនបរិមាណ លើកកម្ពស់គុណភាព និងពង្រឹងសមធម៌នៃធនធានសិក្សាជាខេមរភាសា ជូននិស្សិតដែលកំពុងបន្តការសិក្សា និងត្រៀមខ្លួនធ្វើការស្រាវជ្រាវនៅកម្រិតឧត្តមសិក្សា។ សៀវភៅសិក្សាជាផ្នែកមួយនៃការទទួលស្គាល់គុណភាពអប់រំនៃគ្រឹះស្ថានឧត្តមសិក្សា និងជាធនធានសិក្សាដែលជាមូលដ្ឋានមួយដ៏សំខាន់ ក្នុងការគាំទ្រដល់ការបង្រៀន និងរៀន ហើយត្រូវមានបរិមាណគ្រប់គ្រាន់ ឆ្លើយតបទៅនឹងកម្មវិធីអប់រំ និងតម្រូវការសិក្សាស្រាវជ្រាវ។ ជាគោលការណ៍ គ្រឹះស្ថានឧត្តមសិក្សាទាំងអស់ ត្រូវមានសៀវភៅសិក្សាដែលប្រើជាគោលសម្រាប់មុខវិជ្ជានីមួយៗ។ ចំនួនសៀវភៅសិក្សាដែលគ្រប់គ្រាន់សម្រាប់ការស្រាវជ្រាវ និងការសិក្សារបស់និស្សិត ត្រូវមានយ៉ាងតិចមួយចំណងជើងក្នុងមួយមុខវិជ្ជា ហើយត្រូវតែម្កល់យ៉ាងតិច២ច្បាប់នៅក្នុងបណ្ណាល័យ ឬអាចរកបានតាមប្រព័ន្ធអេឡិចត្រូនិក។ ក្រសួងអប់រំ យុវជន និងកីឡា លើកទឹកចិត្តបន្ថែមទៀតជូនដល់គ្រឹះស្ថានឧត្តមសិក្សារដ្ឋ និងឯកជនដែលបានស្នើសុំថវិកាមូលនិធិ ស.គ.ន រួច សូមចូលរួមបន្ថែមទៀតដើម្បីបង្កើនចំនួនចំណងជើងសៀវភៅ។ ចំណែកគ្រឹះស្ថានឧត្តមសិក្សារដ្ឋ និងឯកជនដែលពុំទាន់បានដាក់ពាក្យស្នើសុំថវិកាមូលនិធិ ដើម្បីរៀបរៀង និពន្ធ និងកែលម្អ សៀវភៅសិក្សានៅកម្រិតឧត្តមសិក្សា សូមរួសរាន់ចូលរួមដើម្បីជាគុណប្រយោជន៍ដល់តម្រូវការដ៏ទទូចនិងថ្លៃថ្លានៃនិស្សិតកម្ពុជាក្នុងការសិក្សា និងស្រាវជ្រាវនៅកម្រិតឧត្តមសិក្សា។

សេចក្តីបញ្ជាក់
នៃមូលនិធិស្រាវជ្រាវ គំនិតច្នៃប្រឌិត និងនវានុវត្តន៍

សៀវភៅសិក្សានេះជាលទ្ធផលនៃការស្នើសុំអនុវត្តថវិកាមូលនិធិស្រាវជ្រាវ គំនិតច្នៃប្រឌិត និងនវានុវត្តន៍ ក្នុងគម្រោងរៀបរៀង និពន្ធ និងកែលម្អសៀវភៅសិក្សា ដែលនឹងត្រូវប្រើប្រាស់នៅកម្រិតឧត្តមសិក្សា។ សៀវភៅសិក្សានេះ ត្រូវបានរៀបរៀង និពន្ធ ឬកែលម្អដោយមានការធានាអះអាងថាជាស្នាដៃរបស់អ្នកនិពន្ធផ្ទាល់ និងបានឆ្លងកាត់ត្រួតពិនិត្យ ផ្តល់យោបល់ និងវាយតម្លៃដោយក្រុមប្រឹក្សាអប់រំ ក្រុមប្រឹក្សាស្រាវជ្រាវ ឬក្រុមប្រឹក្សាដែលមានតម្លៃស្មើនៃគ្រឹះស្ថានឧត្តមសិក្សា និងតាមរយៈកិច្ចសន្យាដែលបានធ្វើឡើង និងដែលបានតម្កល់ទុកនៅមូលនិធិស្រាវជ្រាវ គំនិតច្នៃប្រឌិត និងនវានុវត្តន៍។ រាល់ខ្លឹមសារ ការបកស្រាយ ឬរូបភាព ដែលមាននៅក្នុងសៀវភៅនេះ គឺជាជំហរនិងទស្សនៈផ្ទាល់របស់អ្នកនិពន្ធ ហើយពុំឆ្លុះបញ្ចាំង ឬជាតំណាងដល់មូលនិធិស្រាវជ្រាវ គំនិតច្នៃប្រឌិត និងនវានុវត្តន៍ នៃក្រសួងអប់រំ យុវជន និងកីឡាឡើយ។

អារម្ភកថា

ស្នាដៃនិពន្ធ និងរៀបរៀងសៀវភៅគណិតវិទ្យាសម្រាប់កុំព្យូទ័រមួយនេះ គឺយើងបានលើកយកនូវផ្នែកមួយនៃ គណិតវិទ្យា ព្រមទាំងការអនុវត្តមួយចំនួនដែលពាក់ព័ន្ធនឹងការប្រើប្រាស់សម្រាប់ជាគំនិតក្នុងការសរសេរកូដនៅក្នុងកម្មវិធីណាមួយដែលប្រើប្រាស់ដោយកុំព្យូទ័រ ដោយធ្វើឲ្យអ្នកអានបានយល់ និងមានការសម្រេចចិត្តបានយ៉ាងល្អក្នុងការបង្កើតអ្វីមួយផងដែរ។

សៀវភៅគណិតវិទ្យាកុំព្យូទ័រមួយនេះរៀបចំចងក្រងឡើងដើម្បីជួយសម្រួលដល់និស្សិត ដែលកំពុង និងបានសិក្សាផ្នែកវិទ្យាសាស្រ្តកុំព្យូទ័រ ព្រោះថាវាមានឯកសារនានាជាច្រើនសុទ្ធសឹងជាសៀវភៅដែលសរសេរដោយភាសាបរទេសទាំងអស់ ដែលធ្វើឲ្យនិស្សិត ឬអ្នកអានមានការលំបាកក្នុងការអាន និងស្វែងយល់នូវអត្ថន័យមួយចំនួនក្នុងឯកសារទាំងនោះ។

សៀវភៅគណិតវិទ្យាកុំព្យូទ័រមួយនេះ ត្រូវបានរៀបចំឡើងជាខេមរភាសា ដើម្បីបង្កើតភាពងាយស្រួលដល់និស្សិត ឬអ្នកអាន អាចទទួលបាននូវការសិក្សាស្រាវជ្រាវផ្នែកគណិតវិទ្យាសម្រាប់កុំព្យូទ័រ ព្រមទាំងជាមូលដ្ឋានសម្រាប់ឈានទៅបន្តការសិក្សានៅផ្នែកខាងមុខទៀតបានយ៉ាងល្អប្រសើរ លើសពីនេះទៅទៀត សៀវភៅគណិតវិទ្យាកុំព្យូទ័រមួយនេះ ក៏បានបញ្ចូលនូវវិធីសាស្រ្តដោះស្រាយបញ្ហាជាច្រើនដោយប្រើប្រាស់ភាសា C Programing, C++ Programing ផងដែរ ដើម្បីធ្វើការប្រៀបធៀប និងអនុវត្ត។

សៀវភៅគណិតវិទ្យាកុំព្យូទ័រមួយនេះ ពិតប្រាកដណាស់ថានៅមានចន្លោះខ្វះខាត ហើយក៏ពុំទាន់បំពេញសេចក្តីត្រូវការរបស់និស្សិត ឬអ្នកអាន ជាច្រើនទៀតដែរ អាស្រ័យហេតុនេះហើយ ទើបយើងខ្ញុំដែលជាអ្នករៀបចំសៀវភៅគណិតវិទ្យាកុំព្យូទ័រមួយនេះឡើង រង់ចាំទទួលនូវមតិរិះគន់ដើម្បីស្ថាបនា ពីបណ្តាប្រិយមិត្តអ្នកអានទាំងអស់ ដែលបានចំណាយពេលវេលាដ៏មានតម្លៃ និងយកចិត្តទុកដាក់ក្នុងការអាន និងផ្តល់ជាយោបល់មកដល់យើងខ្ញុំដែលជាអ្នករៀបរៀងសៀវភៅគណិតវិទ្យាកុំព្យូទ័រមួយនេះ។

ជាទីបញ្ចប់ ខ្ញុំសូមថ្លែងអំណរគុណជាអតិបរមាដល់គណៈកម្មការត្រួតពិនិត្យ ដែលបានចំណាយពេលវេលាធ្វើការពិនិត្យ និងវាយតម្លៃនូវសៀវភៅគណិតវិទ្យាកុំព្យូទ័រមួយនេះ។

ចំពោះយោបល់ល្អៗរបស់មិត្តអ្នកអានសៀវភៅនេះ នឹងត្រូវបានទទួលការកែលម្អតាមការលើកឡើងដោយក្តីរីករាយ ដើម្បីជាប្រយោជន៍ និងដើម្បីឲ្យសៀវភៅគណិតវិទ្យាកុំព្យូទ័រមួយនេះកាន់តែមានតម្លាភាព និងសុក្រឹតភាពក្នុងការបោះពុម្ពលើកក្រោយទៀត។

ថ្ងៃសុក្រ ១០កើត ខែជេស្ឋ ឆ្នាំឆ្លូវ ត្រីស័ក ព.ស ២៥៦៥

ត្រូវនឹង ថ្ងៃទី២១ ខែឧសភា ឆ្នាំ២០២១

មាតិកា

មេរៀនទី ១ គណិតតក្កវិទ្យា 1

គោលបំណងនៃមេរៀន 1

1.1 សំណើ 1

1.2 ឈ្នាប់តក្កវិទ្យា 2

1.2.1 ឈ្នាប់មិន “ ¯ ” ឬ “ ¬ ” 2

1.2.2 ឈ្នាប់និង “^” 3

1.2.3 ឈ្នាប់ ឬ “∨” 3

1.2.4 ឈ្នាប់នាំឲ្យ “=>” 4

1.2.5 ឈ្នាប់សមមូល “<=>” 5

1.2.6 ឈ្នាប់តក្កសមមូល “≡” 5

1.3 លក្ខណៈ 6

1.3.1 ច្បាប់ Morgan 6

1.3.2 លក្ខណៈត្រឡប់ 6

1.3.3 លក្ខណៈផ្តុំ 6

1.3.4 លក្ខណៈបំបែក 6

1.3.5 លក្ខណៈឆ្លង 7

លំហាត់ 8

មេរៀនទី ២ សំណុំ 9

គោលបំណងនៃមេរៀន 9

២.១ សំណុំ 9
២.១.១. និយមន័យ...9
២.២ សំណុំស្មើគ្នា...10
២.២ សំណុំរង ឬផ្នែកនៃសំណុំ 11
២.២.១ និយមន័យ 11
២.២.២ លក្ខណៈ...12
២.៣ សំណុំទទេ 13
២.៤ សំណុំរងបំពេញ 14
២.៤.១ លក្ខណៈ 15
២.៥ សំណុំផ្នែកនៃ E...15
២.៦ សំណុំមិនទាក់ទងគ្នា...16
២.៧ ប្រមាណវិធីលើសំណុំ..17
២.៦.១ ប្រសព្វនៃពីរសំណុំ 17
២.៦.២ ប្រជុំនៃសំណុំ 18
លក្ខណៈ 20
លក្ខណៈទំនាក់ទំនងគ្នារវាងប្រសព្វ និងប្រជុំ 20
២.៦.៣ ផលដកនៃសំណុំ...20
លក្ខណៈ 21
២.៧ កាឌីណាល់នៃសំណុំរាប់អស់..21
២.៧.១ និយមន័យ 21

លំហាត់ 24
មេរៀនទី ៣ ប្រព័ន្ធរបាប់ (Number System) 25
គោលបំណងនៃមេរៀន 25
៣.១ ប្រព័ន្ធលេខគ្មានទីតាំង 26
៣.២ ប្រព័ន្ធលេខមានទីតាំង 26
៣.៣ ប្រព័ន្ធគោលពីរ (Binary Number System) 30
៣.៤ ប្រព័ន្ធគោលប្រាំបី (Octal Number System) 32
៣.៥ ប្រព័ន្ធគោលដប់ប្រាំមួយ (Hexadecimal Number System) 33
៣.៦ របៀបបំលែងប្រព័ន្ធគោលនីមួយៗទៅជាប្រព័ន្ធគោលផ្សេងៗ 33
៣.៦.១ វិធីបំលែងពីប្រព័ន្ធគោលពីរទៅជាប្រព័ន្ធគោលដប់ 34
៣.៦.២ វិធីបំលែងពីប្រព័ន្ធគោលប្រាំបីទៅជាប្រព័ន្ធគោលដប់ 35
៣.៦.៣ វិធីបំលែងពីប្រព័ន្ធគោលដប់ប្រាំមួយទៅជាប្រព័ន្ធគោលដប់ 36
៣.៧ របៀបបំលែងប្រព័ន្ធគោលដប់ ទៅជាប្រព័ន្ធគោលថ្មីទៀត 37
៣.៨ របៀបបំលែងប្រព័ន្ធគោលពីរ ទៅជាប្រព័ន្ធគោលប្រាំបី 39
៣.៩ របៀបបំលែងប្រព័ន្ធគោលប្រាំបី ទៅជាប្រព័ន្ធគោលពីរ 40
៣.១០ របៀបបំលែងប្រព័ន្ធគោលដប់ប្រាំមួយ ទៅជាប្រព័ន្ធគោលពីរ 41
៣.១១ របៀបបំលែងប្រព័ន្ធគោលពីរ ទៅជាប្រព័ន្ធគោលដប់ប្រាំមួយ 42
៣.១២ របៀបបំលែងប្រព័ន្ធគោលប្រាំបី ទៅជាប្រព័ន្ធគោលដប់ប្រាំមួយ 42
៣.១៣ របៀបបំលែងប្រព័ន្ធគោលដប់ប្រាំមួយ ទៅជាប្រព័ន្ធគោលប្រាំបី 43
៣.១៤ ចំនួនទសភាគ (fraction number) 45

៣.១៤.១ បំលែងតម្លៃទសភាគ ពីប្រព័ន្ធគោលដប់ ទៅជាប្រព័ន្ធគោលពីរ 46
៣.១៤.២ បំលែងតម្លៃទសភាគ ពីប្រព័ន្ធគោល១០ ទៅជាប្រព័ន្ធគោល៨....................................47
៣.១៤.៣ បំលែងតម្លៃទសភាគ ពីប្រព័ន្ធគោល១០ ទៅជាប្រព័ន្ធគោល១៦ 47
លំហាត់ 49
មេរៀនទី ៤ ជំហាននៃការដោះស្រាយបញ្ហា 50
គោលបំណងនៃមេរៀន 50
៤.១ ក្បួនដោះស្រាយបញ្ហា (Algorithm)..50
៤.១.១ ដំណោះស្រាយបញ្ហាទៅតាមលំដាប់លំដោយ (Sequence Algorithms).........................51
៤.១.២ ក្បួនដោះស្រាយបញ្ហាមានលក្ខខណ្ឌមួយ (If…Statement)..53
៤.១.៣ ក្បួនដោះស្រាយបញ្ហាមានលក្ខខណ្ឌពីរ (if…else statement).......................................55
៤.១.៤ ក្បួនដោះស្រាយបញ្ហាមានដែលមានលក្ខខណ្ឌច្រើនជាងពីរ (if… else-if…).......................57
៤.១.៥ ក្បួនដោះស្រាយបញ្ហាដដែលៗ..60
៤.២ ដំណោះស្រាយបញ្ហាជាគំនូសតាង (Flowchart)..62
៤.២.១ គំនូសតាងជាលំហូរការងារ 63
៤.២.២ គំនូសតាងជាលក្ខណៈមានលក្ខខណ្ឌ..64
៤.២.៣ គំនូសតាងជាលក្ខណៈមានភាពច្រំដែល ឬធ្វើម្តងហើយម្តងទៀត 65
៤.៣ ដំណោះស្រាយបញ្ហាជាភាសាអង់គ្លេស (Pseudo code design language) 72
៤.៣.១ SEQUENCE Pseudo Code 73
៤.៣.២ IF-THEN-ELSE Pseudo Code...74
៤.៣.៣ WHILE Pseudo Code...74

៤.៣.៤ Case Pseudo Code 76
៤.៣.៥ REPEAT-UNTIL Pseudo Code 77
៤.៣.៦ FOR Pseudo Code 78
៤.៣.៧ NESTED CONSTRUCTS Pseudo Code 78
៤.៣.៨ INVOKING SUBPROCEDURES Pseudo Code 79
៤.៣.៩ EXCEPTION HANDLING Pseudo Code 79
លំហាត់ 81
មេរៀនទី៥ ពិជគណិតពិត-មិនពិត 82
និមិត្តសញ្ញានៃ Logic 82
៥.១ តារាងភាពពិតនៃអនុគមន៍ 84
៥.២ អនុគមន៍ Logic ដំបូង 84
៥.២.១ អនុគមន៍ Logic និង (AND) 84
៥.២.២ អនុគមន៍ Logic ឬ or 85
៥.២.៣ អនុគមន៍ logic មិន 86
៥.៣ លក្ខណៈនៃអនុគមន៍ Boolean 86
៥.៤ គោលការណ៍ទ្វេភាពរបស់ Boolean Algebra 87
៥.៥ ការបង្រួមអនុគមន៍ Boolean 88
៥.៦ ធាតុបំពេញនៃអនុគមន៍ Boolean 88
៥.៧ ទម្រង់ Minterm និង Maxterm 89
៥.៨ ផលបូកនៃផលគុណ 89

៥.៩ ផលគុណនៃផលផលបូក 90
៥.១០ Logic Circuit (gate)..90
៥.១០.១ NOT gate 91
៥.១០.២ AND gate 91
៥.១០.៣ OR gate 92
៥.១០.៤ NAND gate..93
៥.១០.៥ NOR gate 93
៥.១១ បណ្តុំនៃ NAND gate..94
៥.១២ បណ្តុំនៃ NOR gate..95
លំហាត់..96
ឯកសារយោង..98

មេរៀនទី ១ គណិតតក្កវិទ្យា

- តើអ្នកធ្លាប់ដែលធ្លាប់បានឮគេនិយាយថាឡូហ្សិក (Logic) ឬច្បាស់អត់ ដែរឬទេ?
- តើហេតុអ្វីបានជាយើងចាំបាច់ត្រូវសិក្សាពី សំណើ តក្កវិទ្យា និងឈ្នាប់តក្កវិទ្យា?

រាល់ទង្វើផ្សេងៗ សុទ្ធតែមានដំណោះស្រាយ ឬអំណះអំណាងជានិច្ច គ្រាន់តែយើងចាំបាច់ត្រូវធ្វើការស្វែងយល់ និងត្រិះរិះនូវសំណើ ឬរឿងនោះជាមុនសិនមុននឹងសម្រេចចិត្ត។ អ្នកចាំបាច់នោះត្រូវយល់ឲ្យច្បាស់ និងរកនូវលទ្ធផលសមស្រប ឬហេតុផលសមស្របនឹងសំណើនោះផងដែរ។ នៅសំណើនីមួយៗ តែងតែមានការស្នើសុំ និងការបង្ហាញលទ្ធផល ប្រកបដោយអំណះអំណាងជានិច្ចដែលយើងនឹងធ្វើការវិភាគមុននឹងសម្រេចចិត្តថាវាអាចកើតមានឡើង ឬមិនអាច និងពិត ឬមិនពិត។

គោលបំណងនៃមេរៀន[1]

នៅក្នុងមេរៀននេះ នឹងពន្យល់បកស្រាយនូវរាល់កាស្នើសុំពីទង្វើ ឬប្រមាណវិធីណាមួយ។

បន្ទាប់ពីអានមេរៀននេះរួច អ្នកនឹងទទួលបាន និងយល់នូវៈ

- យល់ដឹងនូវអ្វីដែលហៅថាសំណើ
- កំណត់ប្រភេទសំណើ
- កំណត់ថាតើតកក្កវិទ្យាជាអ្វី
- បំបែកសំណើ និងផ្គុំសំណើនោះមកវិញ
- បង្ហាញពីឈ្នាប់នៃតក្កវិទ្យា និងលក្ខណៈរបស់វា

1.1 សំណើ

សំណើ គឺជាការប្រជុំពាក្យ ឬនិមិត្តសញ្ញាផ្សេងៗដែលគេធ្វើការអាចអះអាង ឬធ្វើការស្នើដើម្បីចង់ដឹងនូវចម្លើយ និងមានជាលក្ខណៈប្រៀបធៀបនូវចម្លើយនោះ។

ជាទូទៅគេតាងសំណើដោយអក្សរ **p, q, r, t**

ឧទាហរណ៍ p=2 ជាចំនួនគត់ ជាសំណើពិត

q=3^2<2 ជាសំណើមិនពិត

[1] Mathematics for Computer Science revised Monday 18th May, 2015, 01:43.

t=22<3 ជាសំណើមិនពិត

ចំណាំ៖ ចំពោះ "10+1" មិនមែនជាសំណើទេ ព្រោះយើងមិនអាចធ្វើការសម្រេចបានទេថា ពិត ឬ មិនពិតបានទេ ព្រោះនេះជាប្រមាណវិធី មិនមែនជាការប្រៀបធៀបនោះទេ។

-តម្លៃភាពពិតនៃសំណើ

- បើ p ជាសំណើពិតនោះវាមានតម្លៃពិតស្មើ 1។
- បើ q ជាសំណើមិនពិតវាមានតម្លៃមិនពិតស្មើ 0។

* ទ្រឹស្តីបទជាសំណើពិតដោយសន្និដ្ឋានបានតាមការដោះស្រាយបញ្ហា។

* ស្វ័យសត្យជាសំណើពិតជានិច្ចដោយគ្មានការស្រាយបញ្ជាក់។

1.2 ឈ្នាប់តក្កវិទ្យា

- ដូចម្តេចហៅថាឈ្នាប់?

ឈ្នាប់ គឺជាពាក្យសម្រាប់ភ្ជាប់ពីពាក្យមួយទៅពាក្យមួយទៀត ឬពីឃ្លាមួយទៅឃ្លាមួយទៀត ឬ ពីកន្សោមមួយទៅកន្សោមមួយទៀតជាដើម។

- ដូចម្តេចហៅថាឈ្នាប់តក្កវិទ្យា?

ឈ្នាប់តក្កវិទ្យា មិនមែនជាពាក្យសម្រាប់ភ្ជាប់គ្នានោះទេ តែវាគឺជាប្រមាណវិធីមួយបែបដែលប្រើសម្រាប់ភ្ជាប់សំណើពីរ ឬច្រើន។

1.2.1 ឈ្នាប់មិន " ‾ " ឬ " ˥ "

ឈ្នាប់មិន ប្រើសម្រាប់ដាក់មុខសំណើមួយ ហើយសំណើដែលទទួលបាន មានតម្លៃភាពពិតខុសពីសំណើមុខ ឬផ្ទុយពីសំណើដើម។

សំណើ "មិន p" តាងដោយ "˥p" ឬ "$\bar{p}$"

តារាងភាពពិត

p	$\bar{p}$
1	0
0	1

ឧទាហរណ៍ ផ្ទុយនៃសំណើដោយប្រើឈ្នាប់មិន

ទៅលេងឆ្នេរសមុទ្រ ផ្ទុយពីទៅលេង គឺមិនទៅលេង

=> សំណើ "ស្រប=យល់ព្រម" "ស្រប≠មិនស្រប"

1.2.2 ឈ្នាប់និង "^"

ឈ្នាប់ "និង" ប្រើសម្រាប់ភ្ជាប់សំណើពីរ ហើយសំណើ p និង q ជាសំណើពិតលុះត្រាណាតែសំណើទាំងពីរនេះពិតព្រមគ្នា។

ឧទាហរណ៍ៈ "p" ជាត្រីកោណសម័ង្សមានជ្រុង 3 ស្មើគ្នា

"q" ជាត្រីកោណសម័ង្សមានមុំ 3 ស្មើគ្នា

យើងបានសំណើ "p ^ q" ត្រីកោណសម័ង្សមានជ្រុងបី និងមានមុំបី ស្មើគ្នា ជាសំណើពិត។

ឧទាហរណ៍ៈ "p" 15 ចែកដាច់នឹង 5

"q" 15 ចែកដាច់នឹង 3

នាំឲ្យ សំណើ "p ^ q" ជាសំណើពិត។

តារាងភាពពិត

p	q	p ^ q
1	1	1
1	0	0
0	1	0
0	0	0

ចំណាំៈ ឈ្នាប់ "និង" ជាសំណើដែលមិនពិតគ្រប់ករណី លើកលែងតែ "p ^ q" ពិតដូចគ្នា។ ឬអាចនិយាយម្យ៉ាងទៀតបានថាមានន័យជាប្រមាណវិធីគុណ។

1.2.3 ឈ្នាប់ ឬ "v"

សំណើ "p v q" ជាសំណើពិតជានិច្ច លើកលែងតែសំណើ p និង q មិនពិតព្រមគ្នា។

ឧទាហរណ៍ៈ សំណើ p "2<3" ពិត ; សំណើ q " 2=3 " មិនពិត

– p ∨ q "2<=3" ជាសំណើពិត

តារាងភាពពិត

p	q	p ∨ q
1	1	1
1	0	1
0	1	1
0	0	0

1.2.4 ឈ្នាប់នាំឲ្យ "=>"

សំណើ p => q ជាសំណើពិតគ្រប់ករណី តែបើសំណើ p ពិត ហើយ q មិនពិត នោះនាំឲ្យមិនពិត មានន័យថា សំណើខាងមុខពិត ខាងក្រោយមិនពិត គឺទាញទៅរក **មិនពិត** (ល្អ ទៅជួបអាក្រក់ ច្បាស់ណាស់ គឺទៅជាអាក្រក់វិញហើយ)។

ឧទាហរណ៍ៈ p "5>9" ; q "5<9"

យើងបាន (p=>q) ⇔ (5 <> 9) ជាសំណើពិត

ចំណាំៈ សញ្ញា <> អានថាមិនស្មើ

តារាងភាពពិត

p	q	p => q
1	1	1
1	0	0
0	1	1
0	0	1

1.2.5 ឈ្នាប់សមមូល "<=>"

សំណើ p <=> q ជាសំណើពិតក្នុងករណី "p=q" មានន័យថា p និង q ជាសំណើពិតព្រមគ្នា ឬពិតព្រមគ្នា បើនិយាយឲ្យងាយស្រួលចងចាំ គឺបានន័យថាទាល់តែសំណើទាំងនោះមានតម្លៃស្មើគ្នា។

តារាងភាពពិត

p	q	p <=> q
1	1	1
1	0	0
0	1	0
0	0	1

ឧទាហរណ៍៖: p "3>5" មិនពិត q " 8>9" មិនពិត

យើងបាន (p <=> q) ⇔ (3>5 <=> 8>9) ជាសំណើពិត

1.2.6 ឈ្នាប់តក្កសមមូល "≡"

ឈ្នាប់តក្កសមមូល ជាឈ្នាប់មួយដែលគេប្រើប្រាស់សម្រាប់ភ្ជាប់នូវសំណើ ពីរ ឬច្រើន ដែលលទ្ធផលនឹងពិត លុះត្រាតែសំណើទាំងអស់នោះពិតព្រមគ្នា ឬពិតទាំងអស់។

សំណើ p និងសំណើ q ជាសំណើសមាស នោះមានន័យថាសំណើដែលមានទោលជាច្រើន ភ្ជាប់គ្នាដោយ ឈ្នាប់តក្កវិទ្យា។ គេសរសេរ : p≡ q ជាសំណើពិត កាលណា p និង q ពិតដូចគ្នា។

តារាងភាពពិត

p	q	p ≡ q
1	1	1
1	0	0
0	1	0
0	0	0

ឧទាហរណ៍៖: p " 3 < 8" សំណើពិត q "8 > 4" សំណើពិត

យើងបាន (p ≡ q) <=> (3 < 8 & 8 > 4) ជាសំណើពិត

1.3 លក្ខណៈ[2]

1) $\bar{\bar{p}}$ = p

2) (p => q) ≡ $(\bar{q} => \bar{p})$

3) (p=>q) ≡ $(\bar{p} \vee q)$

4) $(\overline{p => q}) \equiv (\bar{p} \wedge \bar{q})$

1.3.1 ច្បាប់ Morgan

$\overline{p \wedge q} = \bar{p} \vee \bar{q}$

$\overline{p \vee q} = \bar{p}^{\bar{q}}$

ជាទូទៅ $\overline{p_1 \wedge p_2 \wedge \dots \wedge p_n} \equiv \overline{p_1 \vee p_2 \vee \dots \vee p_n}$

$\overline{p_1 \vee p_2 \vee \dots \vee p_n} \equiv \overline{p_1 \wedge p_2 \wedge \dots \wedge p_n}$

1.3.2 លក្ខណៈត្រឡប់

p ∨ q ≡ q ∨ p

p ^ q ≡ q ^ p

1.3.3 លក្ខណៈផ្តុំ

p ^ (q ^ r) ≡ (p ^ q) ^ r

p ∨ (q ∨ r) ≡ (p ∨ q) ∨ r

1.3.4 លក្ខណៈបំបែក

p ^ (q ∨ r) ≡ (p ∨ q) ∨ (p ∨ r)

[2] Mathematics for Computer Science revised Monday 18th May, 2015, 01:43.

$p \vee (q \vee r) \equiv (p \wedge q) \vee (p \wedge r)$

1.3.5 លក្ខណៈឆ្លង[3]

$[(p \Rightarrow q) \wedge (q \Rightarrow r)] \equiv (p \Rightarrow r)$

$[(p \Leftrightarrow q) \wedge (q < r)] \equiv (p \Leftrightarrow r)$

ចំណាំ៖ នៅក្នុងសំណើ "p=>q"

- p ហៅថាសម្មតិកម្ម
- q ហៅថាសន្និដ្ឋាន ឬ លក្ខខណ្ឌចាំបាច់

ក្នុងការបេកស្រាយបញ្ជាក់សំណើ "p => q"

សន្មតថាសំណើ p ជាសំណើពិត ហើយមានន័យថាសំណើ p => q ជាសំណើពិត។

[3] Mathematics for Computer Science revised Monday 18th May, 2015, 01:43

លំហាត់

1. ឱ្យ p ពិត, q មិនពិត និង r ពិត ចូររកតម្លៃភាពពិតនៃសំណើខាងក្រោម៖
   a. (p ^ q) => r
   b. (p ^{v} q) => $\bar{r}$
   c. P ^ (q => r)
   d. p => (q => r)
   e. p ^{v} q
   f. ($\bar{p}$ ^{v} (p → (q ^ p))
   g. ($\bar{p}$→ (q ≡ p))
   h. (q ^{v} p) => r)
   i. (p ^ $\bar{p}$) ∨ (q → r)
   j. p => $\bar{r}$
2. ឱ្យ A, B, and C ជាសំណើ ចូរបង្ហាញថា៖
   a. [(A ^ B) => C] ≡ A => B => C
   b. (A => (B∧ C)) ≡ (A ∧ B) ^ (A => C)
   c. (A ∨ B) => C) ≡ (A => C) ^ (B=>C)
   d. [A => (B ∨ C)] ≡ [(A ^ B) =>C]
3. ចូរសរសេរតារាងភាពពិតសម្រាប់សំណើខាងក្រោម៖
   a. (p ∨ q) ^ $\bar{p}$
   b. (p ^ q) ∨ ($\bar{p}$ ∨ q)
   c. ($\overline{p \wedge q}$) ∨ ($\bar{p}$ ∨ r)
   d. ($\bar{p}$ ∨ $\bar{q}$) ∨ p
   e. ($\overline{p \wedge q}$) ∨ (r ^ $\bar{p}$)
   f. (p ∨ q) ∧ ($\overline{r\ v\ q}$).

មេរៀនទី ២ សំណុំ

- តើអ្នកធ្លាប់ដែលបានស្គាល់ ឬធ្លាប់សិក្សាអំពីការប្រមូលផ្តុំដែរទេ ?
- តើហេតុអ្វីបានជាយើងចាំបាច់ត្រូវសិក្សាពី សំណុំ ?

រាល់ការងារជាមួយនឹងកុំព្យូទ័រ ឬគ្រប់គ្រងឯកសារ អ្នកចាំបាច់ត្រូវមានចំណេះដឹងថាតើ អ្នកគួរគ្រប់គ្រងតាមវិធីសាស្ត្រណា ឬរបៀបណា។ ក្នុងការសិក្សានេះ យើងក៏នឹងលើកយកនូវការគ្រប់គ្រង ដែលអ្នកគួរតែដឹងដើម្បីយកទៅប្រើនៅពេលខាងមុខ។

គោលបំណងនៃមេរៀន[4]

នៅក្នុងមេរៀននេះ នឹងពន្យល់បកស្រាយការប្រមូលផ្តុំដែលមានលក្ខណៈជាការផ្តុំបែបដាច់ដោយឡែកពីគ្នា ការផ្តុំជាមួយគ្នាជាដើម។

បន្ទាប់ពីអានមេរៀននេះរួច អ្នកនឹងទទួលបាន និងយល់នូវៈ

- កំណត់ថាតើសំណុំមានអត្ថន័យយ៉ាងដូចម្តេច
- របៀបបង្កើត និងកំណត់ធាតុនៃសំណុំ
- ប្រមាណវិធីនៃសំណុំទាំងនោះ

២.១ សំណុំ

២.១.១. និយមន័យ

សំណុំគឺជាការប្រមូលផ្តុំ ឬ ប្រជុំវត្ថុ (មានជីវិត ឬគ្មានជីវិត) ទាំងឡាយដែលគេតម្រូវឲ្យមានលក្ខណៈរួមមួយ ហៅថា ធាតុនៃសំណុំ។

គេតាងឈ្មោះនៃសំណុំដោយអក្សរពុម្ព A, B, C, ...។

គេតាងធាតុនៃសំណុំដោយអក្សរធម្មតា a, b, c, ...។

- បើ a ជាធាតុនៃសំណុំ E គេសរសេរ "$a \in E$" អានថា a ជារបស់សំណុំ E ឬ "$E \ni a$" សំណុំ E មានធាតុ a នៅក្នុងនោះ
- បើ a ជាធាតុមួយ ហើយស្ថិតក្នុងសំណុំរងនៃ E នោះគេបាន $a \in E$ ពិត

[4] https://ncert.nic.in/ncerts/l/kemh101.pdf

ការកំណត់សំណុំ គេកំណត់តាមពីររបៀប៖

- កំណត់ឈ្មោះសំណុំ
- លក្ខណៈធាតុរបស់សំណុំ ត្រូវសរសេរធាតុទាំងនោះ នៅក្នុង {...}។

ឧទាហរណ៍៖

1. $E = \{a,\ b,\ c\}$
2. $F = \{0, 1, 2 \ldots\}$
3. $M = \{x \in R \ / \ x^2 - x = 0\} = \{0, 1\}$
4. $N = \{x \in N \ / \ x$ ជាចំនួនបឋមតូចជាង $20\} = \{2,3,5,7,11,13,17,19\}$

២.២ សំណុំស្មើគ្នា

សំណុំស្មើគ្នា គឺសំដៅទៅសំណុំទាំងឡាយណាដែលមានធាតុស្មើគ្នា និងដូចគ្នា។ សន្មត់ថាគេមានសំណុំមួយ និងសំណុំមួយទៀត ជាសំណុំស្មើគ្នា គឺត្រូវបានគេសរសេរ A = B ព្រោះ សំណុំទាំងនោះមានធាតុដូចគ្នា និងចំនួនធាតុស្មើគ្នា។

ឧទាហរណ៍៖ A={1, 2, 3, 4, 5}, B={6, 7, 8, 9, 10} ***=> A=B***

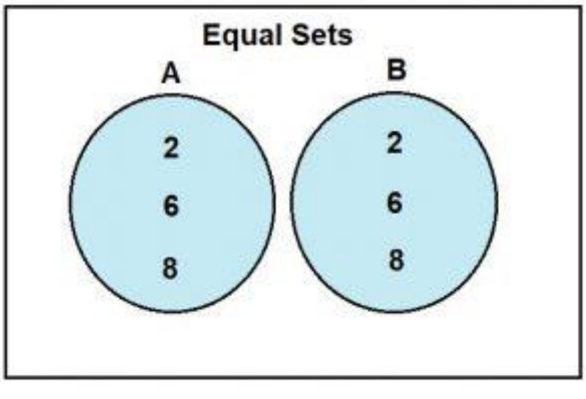
ថ្វីត្បិតថាសំណុំនៅក្នុងឧទាហរណ៍ខាងលើ មានធាតុមិនដូចគ្នាពិតមែន ប៉ុន្តែវាមានចំនួនធាតុស្មើគ្នា នោះវានឹងក្លាយជាសំណុំគ្នា។

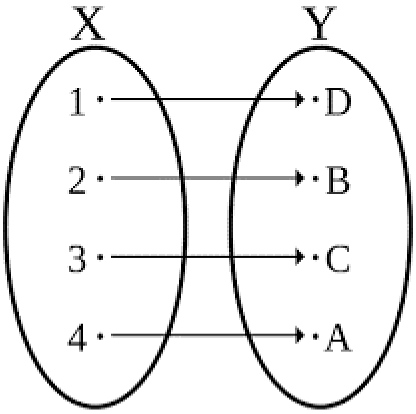
ម្យ៉ាងវិញទៀត មិនថាតែសំណុំដែលមានធាតុប្រភេទជាលេខ ឬជាអក្សរ នោះទេ ឲ្យតែមានចំនួនធាតុនៅក្នុងសំណុំនោះស្មើគ្នា គឺគេចាត់ទុក និងសន្មត់ថា ***សំណុំទាំងនោះជាសំណុំស្មើគ្នា***

$$A = B \iff \forall x\,,\ x \in A \iff x \in B$$

ខាងលើនេះអាន និងមានន័យថា សំណុំ A ស្មើ សំណុំ B សមមូលនឹង គ្រប់តម្លៃ x ព្រោះគ្រប់តម្លៃ x របស់សំណុំ A វាសមមូលនឹងគ្រប់តម្លៃ x នៃសំណុំ B

ដ្យាក្រាមវិន





២.២ សំណុំរង ឬផ្នែកនៃសំណុំ[5]

២.២.១ និយមន័យ

សំណុំរង ឬផ្នែកនៃសំណុំ គឺមានន័យថាសំណុំនោះជាសំណុំដែលមានធាតុទាំងអស់សុទ្ធតែជាធាតុរបស់សំណុំសកល តែសំណុំមិនអាចមានធាតុស្មើនឹងធាតុនៃសំណុំសកលនោះទេ។

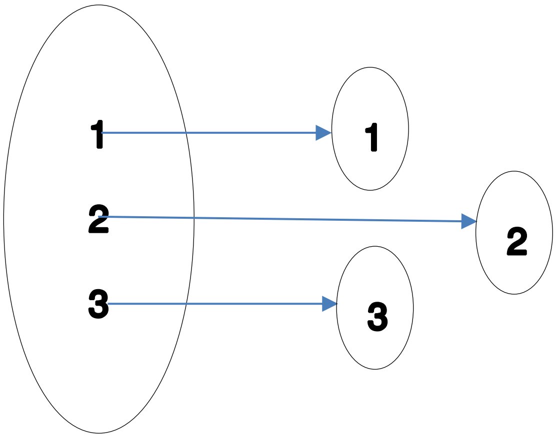
ក្នុងអត្ថន័យនេះផងដែរ យើងអាចនិយាយបានម្យ៉ាងទៀតថា ធាតុដែលមាននៅក្នុងសំណុំសកលគឺអាចយកមកបំបែកទៅជាសំណុំផ្សេងទៀតបាន។

- បើ E ជាសំណុំមួយ ហើយ F ជាសំណុំមួយទៀត គេថា F ជាសំណុំរង ឬផ្នែកនៃសំណុំ E កាលណាសំណុំ F មានធាតុទាំងអស់សុទ្ធតែជាធាតុនៃ E ។

គេសរសេរ៖

$F \subset E$ អានថាសំណុំ F នៅក្នុងសំណុំ E
ឬ $E \supset F$ អានថាសំណុំ E មានសំណុំ F នៅក្នុង

ដ្យាក្រាមវិន



[5] https://en.wikipedia.org/wiki/Complement_(set_theory)

តាមនិយមន័យគេបាន៖

$$F \subset E \Leftrightarrow \forall x,\ x \in F \Rightarrow x \in E$$

ខាងលើនេះអាន និងមានន័យថា សំណុំ F ក្នុងសំណុំសកល E សមមូលនឹងគ្រប់តម្លៃ x ព្រោះតម្លៃ x នៃសំណុំ F នាំទៅរកតម្លៃធាតុ x ទាំងអស់នៃសំណុំសាកល E

ឧទាហរណ៍ទី ១៖ គេឲ្យសំណុំ $E = \{a, b, c, d\}$

គេបាន៖

$\{a\}$, $\{b\}$, $\{c\}$, $\{d\}$, $\{a, b\}$, $\{b, c\}$, $\{a, b, c\}$ សុទ្ធតែជាសំណុំរងនៃ E។

ឧទាហរណ៍ទី ២៖ គេឲ្យសំណុំ $E = N = \{0,1,2,3 \dots\}$

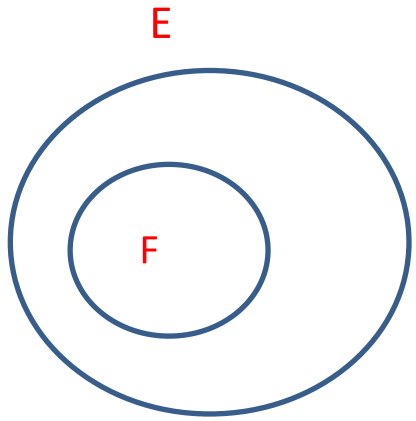
គេបាន៖

សំណុំចំនួនគូ $P = \{0, 2, 4, 6,\dots\}$

សំណុំចំនួនសេស $Q = \{1, 3, 5,\dots\}$

សំណុំពហុគុណនៃ 3 $F = \{0, 3, 6, 9,\dots\}$

⇔ សំណុំ P សំណុំ Q និងសំណុំ F សុទ្ធតែជាសំណុំរងនៃសំណុំសកល E



២.២.២ លក្ខណះ

រាល់ធាតុដែលមាននៅក្នុងសំណុំដើម ឬសំណុំសកលនោះទាំងអស់ គឺត្រូវយកមកសរសេរបំបែកទៅជាសំណុំផ្សេងៗគ្នា ដោយមិនឲ្យច្រំដែល ឬជាន់គ្នា និងមិនអាចមានធាតុស្មើនឹងសំណុំសកលនោះទេ គឺមានន័យថាយកធាតុដែលមាននៅក្នុសំណុំសកលមកបំបែកជាសំណុំផ្សេងទៀតបានទៅតាមធាតុដែលមាននៅក្នុងនោះ។

ឧទាហរណ៍៖ គេមានសំណុំសកលមួយគឺ **E={a, b, c, d}**

ដោយសារនៅក្នុងសំណុំនេះមានធាតុចំនួន ៤ ដូចនេះនាំឲ្យយើងអាចបំបែកសំណុំរងបាន

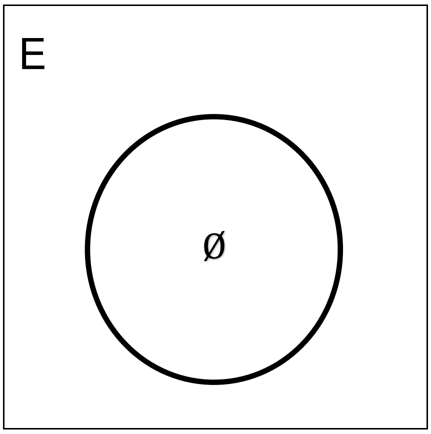
នោះឲ្យគេបាន {a}, {b}, {c}, {d}, {a, b}, {a, c}, {a, d} {a, b, c}, {a, b, d} ជាសំណុំរងនៃ E

២.៣ សំណុំទទេ

សំណុំទទេរតាងដោយ Ø ជាសំណុំដែលគ្មានធាតុ ឬអាចនិយាយបានថា ជាសំណុំដែលមានធាតុទទេរនៅក្នុងនោះ ហើយរាល់សំណើ "x € Ø" ជាសំណើមិនពិត ∀ x ឬគ្រប់ករណី។

Ø = {   }

ដ្យាក្រាមវិន



ឧទាហរណ៍៖: {x € x / x<0} = Ø

2.4.1 ទ្រឹស្តីបទ

ចំពោះគ្រប់សំណុំ E គេបាន E ⊂ E ដែល E ហៅថាផ្នែកពេញនៃ E

ចំពោះគ្រប់សំណុំ E គេបាន ø ⊂ E ដែល ø ហៅថាផ្នែកទទេនៃ E

ស្រាយបញ្ជាក់

E ⊂ E ព្រោះ ∀ x ជាតម្លៃដែលមាននៅក្នុង E នាំឲ្យ x ∈ E ជាសំណើពិត។

ø ⊂ E ព្រោះ ∀ x ដែលមាននៅក្នុង E គឺជាតម្លៃទទេរនៅក្នុងនោះផងដែរ ដែលនាំឲ្យ x ∈ E ជាសំណើពិត។

សម្គាល់៖

បើ $F \subset E$, $F \neq E$, $F \neq \emptyset$ នោះ F ហៅថាសំណុំរងនៃ E

២.៤ សំណុំរងបំពេញ[6]

សំណុំរងបំពេញ គឺជាសំណុំទាំងឡាយណាដែលមានធាតុទាំងអស់ក្នុងសំណុំនោះ មិនមែនជាធាតុរបស់សំណុំរង ប៉ុន្តែធាតុទាំងអស់ដែលមាននៅក្នុងសំណុំនោះ គឺជាធាតុដែលមាននៅក្នុងសំណុំសាកលទាំងអស់ផងដែរ។

E ជាសំណុំសាកលមួយ សំណុំ A ជាសំណុំរងនៃ E ដូចនេះនាំឲ្យសំណុំរងបំពេញនៃ $A \subset E$ គឺជាសំណុំរងនៃ E ដែលមានធាតុទាំងអស់ E តែមិនមែនជាធាតុនៃ A។

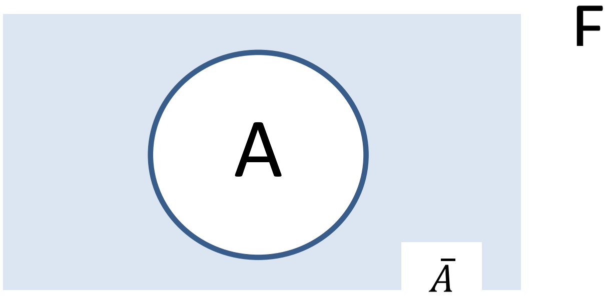
ឧទាហរណ៍៖: E= {a, b, c, d, e, f}

A= {a, b, c}   B= {b, d, e}

$\bar{A}$ = {d, e. f} $\bar{B}$ = {a, c, f}

ពិនិត្យមើលតាមរយៈឧទាហរណ៍ខាងលើ យើងអាចពន្យល់បានថា សំណុំរងបំពេញនៃ A គឺ $\bar{A}$ ព្រោះបើយើងពិនិត្យមើលឡើងវិញ យើងនឹងប្រាកដថា ធាតុដែលមាននៅក្នុងសំណុំ $\bar{A}$ សុទ្ធតែជាធាតុដែលមាននៅក្នុងសំណុំសាកល ហើយក៏មិនមែនជាធាតុដែលមាននៅក្នុងសំណុំរង A នោះទេ។

ដ្យាក្រាមវិន



ដូចនេះ

$$\bar{A} = \{x \in E \,/\, x \notin A\}$$

សំណុំរងបំពេញ $\bar{A}$ មានធាតុជារបស់ E តែមិនមែនជាធាតុរបស់សំណុំរង A

[6] https://ncert.nic.in/ncerts/l/kemh101.pdf

តាមរយៈដ្យាក្រាមនេះបានពន្យល់អ្នកថា គ្រប់តម្លៃរបស់សំណុំរងបំពេញ វាជាតម្លៃរបស់សំណុំសាកល E ដែលតម្លៃជាធាតុរបស់វាមិនមែនជាធាតុរបស់សំណុំរង A។

ឧទាហរណ៍

គេឲ្យសំណុំ $E = \{a, b, c, d, e\}$

គេបាន៖

$A = \{a, c, e\}$ $B = \{a, b, c\}$

$\overline{A} = \{b, d\}$ $\overline{B} = \{d, e\}$

២.៤.១ លក្ខណៈ

$\overline{\overline{A}} = \mathbf{A}$

$\overline{E} = \emptyset$, $\overline{\emptyset} = \mathbf{E}$

២.៥ សំណុំផ្នែកនៃ E

បើ E ជាសំណុំសាកលមួយ នោះសំណុំរងទាំងអស់របស់សាកល E បង្កើតបានជាសំណុំមួយៗហៅថាសំណុំផ្នែកនៃ E ដែលគេតាងដោយ P(E)។

គេអាចបំបែកសំណុំសាកលនោះទៅជាសំណុំផ្នែកបាន តាមរយៈការយកធាតុដែលមាននៅក្នុងសំណុំសាកលនោះមកដាក់ក្នុងសំណុំរងនីមួយៗ ហើយយើងអាចកំណត់ចំនួនសំណុំផ្នែកបានប៉ុន្មានសំណុំ ដោយយកចំនួនធាតុដែលមាននៅក្នុងសំណុំសាកលទាំងអស់មកធ្វើជាស្វ័យគុណលើតម្លៃលេខ ២ នោះយើងនឹងទទួលបានចំនួនសំណុំផ្នែក ឬសំណុំរង ចេញពីសំណុំសាកលនោះ។

ឧទាហរណ៍ ថាគេមានសំណុំសាកលមួយដែលមាន ៣ ធាតុ នោះគេបាន $២^{៣}$ គឺ ៨ សំណុំរង ឬសំណុំផ្នែក។

ឧទាហរណ៍៖:

1. E= ø គេបាន P(E) = {ø}
2. E={a} គេបាន P(E) = {ø, {a}}
3. E= {a, b} គេបាន P(E)= {ø}, {a}, {b}, {a, b}}
4. E={a, b, c} គេបាន P(E)={ ø,{a}, {b}, {c}, {a, b}, {a, c}, {b, c}, {a, b, c}}

កំណត់ចំណាំ៖ កាលណា E ជាសំណុំមួយដែលមាន n ធាតុ នោះ P(E) ជាសំណុំមួយដែលមាន 2^n ធាតុ។

ឧទាហរណ៍៖ នៅក្នុងសំណុំសកល E={1, 2, 3, 4} តើសំណុំនេះអាចបំបែកបានជាប៉ុន្មានផ្នែកនៃសំណុំសកលនេះ?

ដោយហេតុថា នៅក្នុងសំណុំសកលគឺមានធាតុចំនួន ៤ ដូចនេះ ផ្នែកនៃសំណុំ ត្រូវបានកំណត់ថាមាន $2^4 = 16$ ផ្នែកនៃសំណុំ។

២.៦ សំណុំមិនទាក់ទងគ្នា

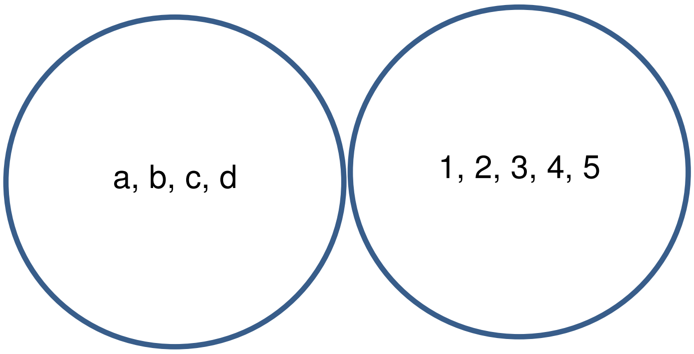
សំណុំដាច់ដោយឡែក ឬសំណុំមិននៅក្នុងសំណុំ គឺជាប្រភេទសំណុំទាំងឡាយណាដែលមានធាតុផ្សេងគ្នា គ្មានតម្លៃនៅក្នុងសំណុំណាមួយនោះទេ។

បើយើងនិយាយថា P មិនមែនជាសំណុំរងនៃ E ទនោះរាល់ធាតុទាំងអស់នៃ P គឺមិនមាននៅក្នុង E ឬអាចនិយាយម្យ៉ាងទៀតបានថាធាតុរបស់ P មិនមែនជាធាតុរបស់សំណុំ E នោះទេ។

គេអាចសរសេរ៖

$$P \not\subset E$$

ដ្យាក្រាមវិន



តាមរយៈដ្យាក្រាមបានបង្ហាញថា សំណុំមិនទាក់ទងគ្នា ជាសំណុំដែលមិនអាចមានធាតុណាមួយដូចគ្នា ឬមិនស្មើគ្នាតែម្តង។

ឧទាហរណ៍៖ គេមានសំណុំ A={1, 2, 3, 4} សំណុំ E={a, b, c, e, f} ដូចនេះទំនាក់ទំនងរវាងសំណុំទាំងពីរ គឺជាទំនាក់ទំនង សំណុំមិនទាក់ទងគ្នា ព្រោះរាល់ធាតុនីមួយៗនៃសំណុំទាំងពីរ គឺមិនដូចគ្នាទាល់តែសោះសូម្បីមួយធាតុ។

២.៧ ប្រមាណវិធីលើសំណុំ[7]

នៅក្នុងការសិក្សានេះ គេនឹងយកសំណុំទាំងនោះមកធ្វើប្រមាណវិធីជាមួយគ្នា ដូចគ្នានឹងការគណនាតម្លៃលេខផងដែរ គ្រាន់តែវាមានភាពផ្សេងគ្នាខ្លះត្រង់ថាមិនមែនយកសំណុំនោះទៅគុណ ឬចែកបាននោះទេ គឺបានត្រឹមតែយកមក បូក ដក ប៉ុណ្ណោះ។

ក្នុងផ្នែកមួយនេះគេយកសំណុំ E ជាសំណុំសកល ហើយសំណុំដទៃទៀតមានដូចជា A, B, C, D ជាសំណុំរងនៃ E ទាំងអស់។

ប្រមាណវិធីនៃសំណុំដែលយើងនឹងសិក្សារួមមាន ប្រសព្វនៃសំណុំ ប្រជុំនៃសំណុំ និងផលដកនៃសំណុំ។

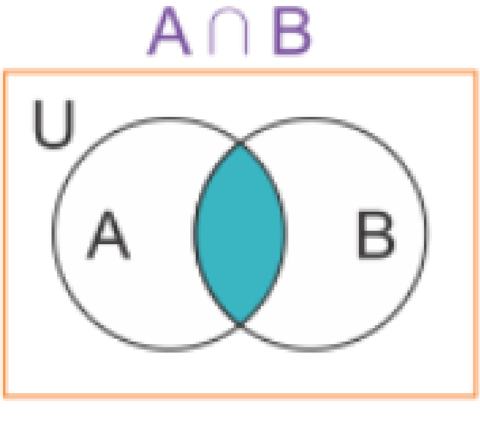
២.៦.១ ប្រសព្វនៃពីរសំណុំ

បើ A និង B ជាសំណុំរងនៃ E ប្រសព្វនៃសំណុំពីរ ឬសំណុំច្រើន មានន័យថាសំណុំទាំងឡាយណា ដែលមានធាតុដូចគ្នានៅក្នុងសំណុំទាំងនោះ គឺជាលទ្ធផលនៃប្រសព្វសំណុំនោះ។

បើ A និង B ជាសំណុំរងនៃ E។

គេកំណត់បាន $A \cap B=\{x : x \in A \text{ and } x \in B \}$

ដ្យាក្រាមវិន



តាមរយៈដ្យាក្រាមវិន បានបង្ហាញថាកន្លែងដែលសំណុំមួយបានកាត់គ្នា និងជាប់គ្នានោះ គឺជាកន្លែងប្រសព្វ ដែលអាចនិយាយបានវាមានធាតុដែលដូចគ្នានៅត្រង់នោះ។

[7] https://www.vedantu.com/ncert-solutions/ncert-solutions-class-11-maths-chapter-1-sets

ឧទាហរណ៍៖

A= {1, 2, 3, 4, 5, 6}, B= {4, 5, 6}

A ∩ B = {4, 5, 6} ប្រសព្វនៃពីរសំណុំខាងលើគឺ 4 5 6 ដោយសារថាទាំងពីរសំណុំខាងលើគឺមានធាតុបីនេះដូចគ្នា។។

ឧទារហណ៍៖

គេមានសំណុំ A = { a, b, c, d }, B = {a, c, d }

ដូចនេះគេបាន៖

$A \cap B =$ { a, c, d}

លក្ខណ:

i. $A \cap B = B \cap A$

ii. $A \cap A = A$

iii. $A \cap E = A$

iv. $A \cap \emptyset = \emptyset$

v. $A \cap \bar{A} = \emptyset$

vi. $A \cap B \subset A,\ A \cap B \subset B$

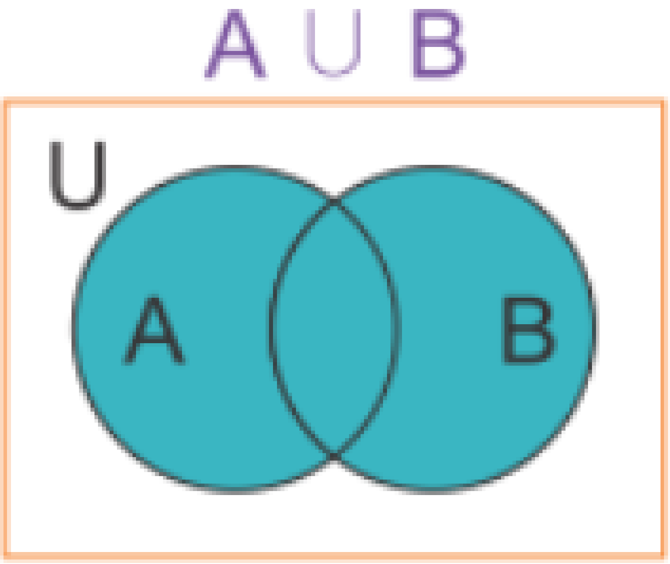
vii. $(A \cap B) \cap C = A \cap (B \cap C)$ គេអាចសរសេរ $A \cap B \cap C$

ជាទូទៅបើ $A_1, A_2, A_3, \ldots A_n$ ជា n សំណុំរងនៃ E នោះគេបាន៖

$$\bigcap_{i}^{n} A_i = A_1 \cap A_2 \cap A_3 \cap \ldots \cap A_{n-2} \cap A_{n-1} \cap A_n$$

២.៦.២ ប្រជុំនៃសំណុំ

ប្រជុំនៃសំណុំពីរ ឬច្រើន មានន័យថាគឺជាធាតុទាំងអស់ដែលមាននៅក្នុងសំណុំទាំងនោះ គឺត្រូវយកមកធ្វើជាលទ្ធផលទាំងអស់ តែធាតុទាំងឡាយណាដែលដូចគ្នា គឺយកមកដាក់តែម្តងប៉ុណ្ណោះ។ កាលណាបើ A, B ជាសំណុំរងនៃ E នាំឱ្យគេបាន A ∪ B= {x ∈ E/x ∈ A v x ∈ B}។



តាមរយៈដ្យាក្រាមវិន បានបង្ហាញថាកន្លែងដែលប្រសព្វគ្នាក៏យក កន្លែងដែលមិនប្រសព្វគ្នាក៏យក ប៉ុន្តែគឺយក តែម្តងប៉ុណ្ណោះ។

ឧទាហរណ៍៖: A={a, b, c, d}, B={a, c, d, f}

A ∪ B= {a, b, c, d, f}

បើ A និង B ជាសំណុំរងនៃ E។

គេកំណត់

$$\boldsymbol{A \cup B = \{x \in E \ / x \in A \vee x \in B\}}$$

អានថា សំណុំ A ប្រជុំសំណុំ B គឺស្មើនឹងគ្រប់តម្លៃ x នៃសំណុំសាកល E ដកគ្រប់តម្លៃ x នៃសំណុំរង A ប្រជុំនឹងតម្លៃ x នៃសំណុំរង B។

<u>ឧទាហរណ៍</u>

គេមានសំណុំ

A = {a, b, c, e }, B = {a, c, d, f }

ដូចនេះគេបាន

$A \cup B = \{a, b, c, d, e, f\}$

លក្ខណៈ

i. $A \cup B = B \cup A$

ii. $A \cup A = A$

iii. $A \cup E = E$

iv. $A \cup \emptyset = A$

v. $A \cup \bar{A} = A$

vi. $A \subset A \cup B$, $B \subset A \cup B$

vii. $(A \cup B) \cup C = A \cup (B \cup C)$ គេអាចសរសេរ $A \cup B \cup C$

ជាទូទៅបើ $A_1, A_2, A_3, \ldots A_n$ ជា n សំណុំរងនៃ E នោះគេបាន៖

$$\bigcup_{i}^{n} A_i = A_1 \cup A_2 \cup A_3 \cup \ldots \cup A_{n-2} \cup A_{n-1} \cup A_n$$

លក្ខណៈទំនាក់ទំនងគ្នារវាងប្រសព្វ និងប្រជុំ

លក្ខណៈបំបែក

$A \cap (B \cup C) = (A \cap B) \cup (A \cap C)$

$A \cup (B \cap C) = (A \cup B) \cap (A \cup C)$

ច្បាប់ Morgan

$\overline{A \cap B} = \bar{A} \cup \bar{B}$

$\overline{A \cup B} = \bar{A} \cap \bar{B}$

២.៦.៣ ផលដកនៃសំណុំ[8]

ផលដកនៃសំណុំពីរ ឬច្រើន មានន័យថារាល់ធាតុទាំងឡាយណាដែលមាននៅក្នុងសំណុំទាំងនោះ គឺត្រូវយកមកផ្ទៀងផ្ទាត់គ្នា កាលណាធាតុដែលមិនដូចគ្នានៃសំណុំខាងមុខ គឺជាលទ្ធផល។

ចំណាំ៖ A \ B ខុសពី B \ A

ស្រាយបញ្ជាក់៖ A={5, 6, 7, 8}, B={7, 8, 9} នាំឲ្យ A \ B={5, 6} តែ B \ A={9}

បើ A, B ជាសំណុំរងនៃ E នោះគេទទួលបាន $A \setminus B = \{x \in E / x \in A \wedge x \notin B\}$។

[8] https://www.vedantu.com/ncert-solutions/ncert-solutions-class-11-maths-chapter-1-sets

B\ A={x ∈ E / x ∈ B ∧ x ∈ A}

បើ A និង B ជាសំណុំរងនៃ E នោះគេបាន៖

$$A \setminus B = \{x \in E / x \in A \wedge x \notin B\}$$

អានថា សំណុំ A ដកសំណុំ B គឺស្មើនឹងគ្រប់តម្លៃ x នៃសំណុំសាកល E ដកគ្រប់តម្លៃ x នៃសំណុំរង A ប្រសព្វនឹងតម្លៃ x មិនមានៅក្នុងសំណុំរង B។

លក្ខណៈ

(i) A\B និង B\A ជាសំណុំពីរដាច់ពីគ្នា (សំណុំពីរដាច់គ្នា កាលណាវាមានប្រសព្វស្មើ ∅)

(ii) $A\setminus B = A\setminus(A \cap B)$

B\A = B\(A∩ B)

(iii) $A\setminus A = \emptyset$

(iv) $A\setminus\emptyset = A$

ឧទាហរណ៍

ឧបមាថា A = { 1, 2, 4, 5, 7, 8 }, B = { 1, 3, 6, 7, 9 }

គេបាន៖

A \ B = { 2, 4, 5, 8 }

B \ A = { 3, 6, 9 }

២.៧ កាឌីណាល់នៃសំណុំរាប់អស់

២.៧.១ និយមន័យ

គេថាសំណុំមួយជាសំណុំរាប់អស់ (ឬកំណត់) (Finite sets) កាលណាសំណុំនេះមាន m ជាតុផ្សេងគ្នាដែល m ជាចំនួនគត់វិជ្ជមាន។ បើមិនដូចនេះ គេថាជាសំណុំមិនកំណត់ (Infinite Sets)។

- សំណុំទទេ គឺជាសំណុំកំណត់
- សំណុំអក្ខរក្រម គឺជាសំណុំកំណត់
- សំណុំនៃចំនួនគត់វិជ្ជមាន A = {2, 4, 6 …} គឺជាសំណុំគ្មានកំណត់។

ឧទាហរណ៍

បើ A ជាសំណុំរាប់អស់ នោះគេតាង n(A) គឺជាចំនួនធាតុនៃសំណុំ A

ទ្រឹស្តីបទ

បើ A និង B ជាសំណុំពីរដាច់ពីគ្នា និងរាប់អស់ នោះគេបាន
A ∪ B ជាសំណុំរាប់អស់។

គេបាន៖

$$n(A \cup B) = n(A) + n(B)$$

ជាទូទៅបើ $A_1, A_2, \ldots, A_n$ ជា n សំណុំដាច់គ្នាពីរៗ និងរាប់អស់
នោះគេបាន៖

$$\boldsymbol{n(A_1 \cup A_2 \cup \ldots \cup A_n) = n(A_1) + n(A_2) + \ldots + n(A_n)}$$

ទ្រឹស្តីបទ

បើ A និង B ជាសំណុំពីរទាក់ទងគ្នា និងរាប់អស់ នោះគេបាន
$A \cup B$ និង $A \cap B$ ជាសំណុំរាប់អស់។

គេបាន៖

$$\boldsymbol{n(A \cup B) = n(A) + n(B) - n(A \cap B)}$$

លំហាត់

១. គេឲ្យ A = { 1, 3, 5, 7, 9 }

B = {1, 5, 6, 7 } និង

C = { 1, 2, 4, 6, 8, 9 }

គណនាសំណុំនីមួយៗដូចខាងក្រោមៈ

ក. $A \cup B$

ខ. $A \cap B$

គ. $(A \cup B) \cap C$

ឃ. $(A \cap B) \cup C$

ង. $(A \cup C) \cap (B \cup C)$

២. គេឲ្យ E = { 0, 1, 2, 3, 4, 5, 6, 7, 8, 9 }

A = {1, 3, 5, 9 }

B = { 2, 4, 6, 8}

C = { 1, 3, 4, 6 }

គណនាសំណុំនីមួយៗដូចខាងក្រោមៈ

ក. $\bar{A}$

ខ. $\overline{(A \cap B)}$

គ. $\bar{A} \cup \bar{B}$

ឃ. $\bar{A} \cap \bar{B}$

ង. $\overline{(A \cup B \cup C)}$

៣. គេមានសំណុំមួយដូចខាក្រោម

G = {d, a, r, e}

តើសំណុំណាមួយខាងក្រោមជាសំណុំរងនៃសំណុំខាងលើ

ក. A = {d, b, c}

ខ. B = {a, e, r}

គ. C = {e, r, f}

មេរៀនទី ៣ ប្រព័ន្ធរាប់ (Number System)

- អ្នកធ្លាប់ដែលធ្លាប់ស្គាល់លេខ ឬតម្លៃលេខពីមុនមកទេ?
- អ្នកធ្លាប់ដែលមានចម្ងល់ថា ហេតុអ្វីបានជាគេអានលេខនោះ ទៅជាបែបនេះ ឬយ៉ាងនេះអ៊ីចឹង?

នៅក្នុងការសិក្សាមេរៀនប្រព័ន្ធរាប់ ឬអាចនិយាយបានដោយសាមញ្ញថាប្រព័ន្ធលេខ នឹងពន្យល់បង្ហាញអ្នកអំពីប្រព័ន្ធលេខដែលមាននៅក្នុងជីវភាពរស់នៅរបស់យើង និងប្រព័ន្ធលេខដែលសម្រាប់ប្រើនៅក្នុងបច្ចេកវិទ្យា និងឌីជីថល នាពេលកន្លងមក និងកំពុងប្រើប្រាស់សព្វថ្ងៃនេះផងដែរ។

គោលបំណងនៃមេរៀន[9]

នៅក្នុងមេរៀននេះ នឹងពន្យល់បកស្រាយនូវរាល់តម្លៃលេខដែលមាននៅក្នុងប្រព័ន្ធគោលនីមួយៗ ព្រមទាំងមានជាឧទាហរណ៍ផងដែរ។

បន្ទាប់ពីអានមេរៀននេះរួច អ្នកនឹងទទួលបាន និងយល់នូវ៖

- យល់ដឹងអំពីប្រព័ន្ធលេខគ្មានទីតាំង
- យល់ដឹងអំពីប្រព័ន្ធលេខមានទីតាំង
- ប្រព័ន្ធគោលពីរ
- ប្រព័ន្ធគោលប្រាំបី
- ប្រព័ន្ធគោលដប់
- ប្រព័ន្ធគោលដប់ប្រាំមួយ
- វិធីសាស្ត្របំលែងពីប្រព័ន្ធគោលដប់ ទៅជាប្រព័ន្ធគោលពីរ
- វិធីសាស្ត្របំលែងពីប្រព័ន្ធគោលដប់ ទៅជាប្រព័ន្ធគោលប្រាំបី
- វិធីសាស្ត្របំលែងពីប្រព័ន្ធគោលដប់ ទៅជាប្រព័ន្ធគោលដប់ប្រាំមួយ
- វិធីសាស្ត្របំលែងពីប្រព័ន្ធគោលពីរ ទៅជាប្រព័ន្ធគោលដប់
- វិធីសាស្ត្របំលែងពីប្រព័ន្ធគោលដប់ ដែលមានក្បៀស ទៅជាប្រព័ន្ធគោលពីរ
- វិធីសាស្ត្របំលែងពីប្រព័ន្ធគោលពីរ ដែលមានក្បៀស ទៅជាប្រព័ន្ធគោលដប់ និងប្រព័ន្ធគោលដទៃទៀតផងដែរ

[9] https://www.gatevidyalay.com/tag/number-system-lecture-notes-pdf/

ប្រព័ន្ធលេខមានលក្ខណៈខុសគ្នាជាច្រើន ដែលប្រហែលមកជាពី សម័យបុរេប្រវត្តិ ឬបុរាណត្រូវបាន បង្កើត និងវិវត្តន៍វាមក។ នៅក្នុងចំណោម ប្រព័ន្ធលេខគោលពីរ គឺជាប្រព័ន្ធលេខមួយបែប ដែលមានភាព សាមញ្ញ និងមានប្រសិទ្ធភាពបំផុត ដែឡត្រូវបានប្រើយ៉ាងទូលំទូលាយក្នុងប្រព័ន្ធឌីជីថល។ ការសិក្សាប្រព័ន្ធ លេខអាចជួយអ្នកឱ្យយល់ពីដំណើរការគណនាមូលដ្ឋានដោយប្រព័ន្ធឌីជីថល។

ប្រព័ន្ធគោលនៅក្នុងប្រព័ន្ធរាប់ គឺគ្រាន់តែជាចំនួនដែលមានតួ ដែលគេសម្គាល់ និងកំណត់ដោយ ពាក្យថាតួ (Digit) ផ្សេងៗគ្នារួមទាំងលេខសូន្យ (0) នៅក្នុងប្រព័ន្ធរាប់ទាំងនោះ។ តួនីមួយៗ ដែលមាននៅ ក្នុងប្រព័ន្ធគោលពីរ ត្រូវបានគេស្គាល់ថាជាលេខ សម្រាប់ប្រើប្រាស់ក្នុងប្រព័ន្ធអេឡិចត្រូនិក ឬអគ្គិសនី ហើយ មានតែមួយក្នុងតម្លៃលេខពីរប៉ុណ្ណោះ គឺ 0 និង ១។

៣.១ ប្រព័ន្ធលេខគ្មានទីតាំង

កាលពីជំនាន់ដើម ឬពីដើមដំបូង មនុស្សយើងគឺមិនបានស្គាល់ ឬប្រើប្រាស់ប្រព័ន្ធរាប់នោះទេ គឺ មនុស្សបានប្រើម្រាមដៃសម្រាប់រាប់ចំនួន ហើយនៅពេលម្រាមដៃទាំងអស់ ហើយគេនៅតែរាប់មិនទាន់អស់ ឬគ្រប់ គេយកវត្ថុផ្សេងៗ ដែលមានទំហំតូចៗ មកបន្ថែម ឬជំនួសម្រាមដៃ ដែលការធ្វើបែបនេះហៅថា ***ប្រព័ន្ធ លេខដែលគ្មានទីតាំង***។ និមិត្តសញ្ញានីមួយៗតាងឱ្យតម្លៃខុសៗគ្នា ដោយមិនគិតពីទីតាំង នៃចំនួនរបស់វាក្នុង ចំនួននិម្មិតសញ្ញា ត្រូវបានគេបន្ថែមយ៉ាងងាយដើម្បីបញ្ជាក់ពីតម្លៃសេសរបស់វានោះទេ។

ដោយហេតុថាវាមានការលំបាកក្នុងការអនុវត្តគណិតលេខ ជាមួយប្រព័ន្ធរាប់ទាំងនោះ អ្នកប្រាជ្ញ ឬ គណិតវិទូ គាត់បានរិះរកនូវវិធីសាស្ត្រមួយចំនួន ដែលនៅក្នុងនោះ គាត់ក៏បានបង្កើតនូវប្រព័ន្ធរាប់ ដែលមាន និម្មិតសញ្ញា និងទីតាំងច្បាស់លាស់ ដែលហៅថា ***ប្រព័ន្ធលេខមានទីតាំង***។

៣.២ ប្រព័ន្ធលេខមានទីតាំង

នៅក្នុងប្រព័ន្ធលេខមានទីតាំង មាននិមិត្តសញ្ញាតែពីរ ឬបីប៉ុណ្ណោះដែលហៅថា **ចំនួនតួ** ហើយនិមិត្ត សញ្ញាទាំងអស់នោះតាងឱ្យតម្លៃលេខអាស្រ័យទៅលើទីតាំងដែលលេខស្ថិតនៅ។

ជាឧទាហរណ៍ ដ៏ល្អ និងសមញ្ញ នៃប្រព័ន្ធលេខមានទីតាំង គឺប្រព័ន្ធលេខគោលដប់ ដែលយើងប្រើវា ស្ទើរតែគ្រប់ទីកន្លែងដែលត្រូវការលេខ។ ឧទាហរណ៍ មួយទៀត គឺប្រព័ន្ធគោលពីរដែលត្រូវបានប្រើ ប្រព័ន្ធ លេខមូលដ្ឋានសម្រាប់កុំព្យូទ័រទាំងអស់។ នៅក្នុងប្រព័ន្ធលេខទីតាំងលេខមួយត្រូវបានតំណាងដោយខ្សែរលេ ខដែលទីតាំងនៃខ្ទង់នីមួយៗត្រូវបានផ្សារភ្ជាប់ជាមួយនឹងទម្ងន់។

ជាទូទៅប្រព័ន្ធលេខមានទីតាំងត្រូវបានបញ្ជាក់ដូចជា៖

- តួខ្លួនឯង
- ទីតាំងរបស់នៃតួនោះនៅក្នុងចំនួនណាមួយ
- ប្រព័ន្ធគោលរបស់ប្រព័ន្ធរាប់នោះ

នៅក្នុងប្រព័ន្ធរាប់ដែលយើងប្រើប្រាស់រាល់ថ្ងៃគេហៅថាប្រព័ន្ធគោលដប់ (Decimal number system) នៅក្នុងប្រព័ន្ធរាប់នេះគោលស្មើ 10 ហើយមាន 10 តួលេខ ដែលគេកំណត់ដូចតទៅ៖

គោលដប់ (0, 1, 2, 3, 4, 5, 6, 7, 8, 9) 10 តួ តាងឱ្យប្រព័ន្ធគោលដប់ យើងដឹងរួចមកហើយថានៅក្នុងប្រព័ន្ធគោលដប់ គេតាងដោយ ខ្ទង់រាយ ខ្ទង់ដប់ ខ្ទង់រយ ខ្ទង់ពាន់ ខ្ទង់ម៉ឺន ខ្ទង់សែន ខ្ទង់លាន។ ប៉ុន្តែយើងប្រហែលជាមិនយកចិត្តទុកដាក់ ឬអើពើអំពីទីតាំងនីមួយៗរបស់វានោះទេ ហើយក៏មិនបានកំណត់ឱ្យស្វ័យគុណនៃគោលនោះដែរ។ ប៉ុន្តែវាត្រូវបានគេកណត់ទីតាំងដាក់លាក់ទៅខាងឆ្វេងនៃឯកតាដែលតាងឱ្យតម្លៃនៃប្រព័ន្ធគោលដប់នោះ។

ឧទាហរណ៍ៈ គេឱ្យតម្លៃលេខនៃប្រព័ន្ធគោលដប់ គឺ 2568 គេសរសេរ $(2568)_{10}$ ហេតុអ្វីបានជាយើងមានលទ្ធភាពក្នុងការអានលេខនោះ ដូចដែលធ្លាប់បានអានពីមុនមកថា ***ពីរពាន់ប្រាំរយហុកសិបប្រាំបី***?

ដូចដែលគេបានកំណត់តួរបស់វាដើម្បីមកបញ្ជាក់ដូចខាងក្រោម៖

8 ស្ថិតនៅតំណាងឱ្យឯកតាខ្ទង់រាយ (unit) គឺតួទី 0

6 ស្ថិតនៅតំណាងឱ្យឯកតាខ្ទង់ដប់ (Ten) គឺតួទី ១

5 ស្ថិតនៅតំណាងឱ្យឯកតាខ្ទង់រយ (Hundred) គឺតួទី ២

2 ស្ថិតនៅតំណាងឱ្យឯកតាខ្ទង់ពាន់ (Thousand) គឺតួទី ៣

ដោយធ្វើការកំណត់តួរបស់លេខនីមួយៗរួចមក គេចាំបាច់យកតម្លៃលេខនីមួយៗនោះ មកលើកជាស្វ័យគុណនៃតួដែលវាស្ថិតនៅ រួចយកមកគុណនឹងប្រព័ន្ធគោលរបស់វា ទើបអ្នកយកតម្លៃលេខដែលទទួលបានពីការគុណនោះ មកបូកបញ្ចូលគ្នាដើម្បីទទួលបានលទ្ធផល។

យើងអាចសរសេរវាបានថា៖

$8x10^0, 6x10^1, 5x10^2, 2x10^3$

សរសេរចំនួន $(2568)_{10} = (2x1000) + (5x100) + (6x10) + (8x1)$

$= 2000 + 500 + 60 + 8$ (ពីរពាន់ ប្រាំរយ ហុកសិប ប្រាំបី)

$(2568)_{10} = (2568)_{10}$

ដូចនេះគេអាចដាក់វាទៅតាមទីតាំងដោយប្រើតួ និងការរៀបចំលេខទាំងនោះឱ្យប្រើក្នុងទីតាំងផ្សេងគ្នា។ គ្រប់ប្រព័ន្ធរាប់នៃគោលគឺតម្លៃសរុបនៃនិមិត្តសញ្ញា (Digit) ផ្សេងៗគ្នាដែលប្រព្រឹត្តទៅនៅក្នុងប្រព័ន្ធរាប់។

ម្យ៉ាងវិញទៀត យើងក៏អាចធ្វើការកំណត់ និងគណនាវាបានដែរប្រសិនបើយើងមាននូវតារាងដែលបានកំណត់ និងប្រៀបធៀបខាងក្រោមនេះ៖

តារាង ប្រព័ន្ធគោលទាំងបួន និងឌីជីថល[10]

Decimal	Binary	Octal	Hexadecimal	Base-32
0	00000	0	0	0
1	00001	1	1	1
2	00010	2	2	2
3	00011	3	3	3
4	00100	4	4	4
5	00101	5	5	5
6	00110	6	6	6
7	00111	7	7	7
8	01000	8	8	8
9	01001	9	9	9
10	01010	12	A	A
11	01011	13	B	B
12	01100	14	C	C
13	01101	15	D	D
14	01110	16	E	E
15	01111	17	F	F
16	10000	20	10	G
17	10001	21	11	H
18	10010	22	12	I
19	10011	23	13	J
20	10100	24	14	K
21	10101	25	15	L
22	10110	26	16	M
23	10111	27	17	N
24	11000	30	18	O

[10] https://byjus.com/maths/number-system-conversion/

25	11001	31	19	P
26	11010	32	1A	Q
27	11011	33	1B	R
28	11100	34	1C	S
29	11101	35	1D	T
30	11110	36	1E	U
31	11111	37	1F	V

ក្រោយពីយើងបានមើលឃើញនូវតារាងខាងលើរួចមក យើងអាចធ្វើការចងចាំនូវតារាងនេះ ដើម្បីធ្វើការងារជាច្រើននៅចំណុចជាបន្តបន្ទាប់។

ឧទាហរណ៍៖ ការបំលែងពីប្រព័ន្ធគោលពីរ ទៅជាប្រព័ន្ធគោលប្រាំបី និងប្រព័ន្ធគោលដប់ប្រាំមួយ ទាំងមានក្បៀស និងគ្មានក្បៀសជាលើកដំបូងដោយការចាំតារាង។

ឧទាហរណ៍៖

1- $(11010110)_2 = (?)_8$

a. 011 = 6

b. 010 = 2

c. 11 → 011 = 3

⇨ $(11010110)_2 = (326)_8$

2- $(11010110)_2 = (?)_{16}$

a. 0110 = 6

b. 1101 = 14 → D

⇨ $(11010110)_2 = (D6)_{16}$

ឧទាហរណ៍៖ ការបំលែងពីប្រព័ន្ធគោលប្រាំបី ទៅជាប្រព័ន្ធគោលពីរ និងគោលដប់ប្រាំមួយ ដោយការចាំតារាង

1- $(273)_8 = (?)_2 = (?)_{16}$

a. $(2)_8 = 010$

b. $(7)_8 = 111$

c. $(3)_8 = 011$

⇨ **$(273)_8 = (010111011)_2$**

ក្រោយពីទទួលបានលទ្ធផលបន្ទាប់ពីបំបែកតម្លៃលេខពីប្រព័ន្ធគោលប្រាំបី ទៅជាប្រព័ន្ធគោលពីររួចហើយ យើងគឺប្រាកដណាស់ថាអាចបំបែកបន្ត ពីប្រព័ន្ធគោលពីរ ទៅជាប្រព័ន្ធគោលដប់ប្រាំមួយបានយ៉ាងងាយ ដោយគ្រាន់តែយកតម្លៃលេខប្រព័ន្ធគោលពីរដែលបានបំបែករួច មករៀបជាបួនតួ ហើយបំបែកទៅជាតម្លៃលេខដែលមាននៅក្នុងប្រព័ន្ធគោលដប់ប្រាំមួយជាការស្រេច។

1- $(273)_8 = (?)_2 = (?)_{16}$

a. $(2)_8 = 010$

b. $(7)_8 = 111$

c. $(3)_8 = 011$

⇨ $(273)_8 = (010111011)_2 = (?)_{16}$

- 1011 = 11 = B
- 1011 = 11 = B

⇨ **$(273)_8 = (010111011)_2 = (BB)_{16}$**

ការជ្រើសរើសជាលើកដំបូងនៃចំនួនតួ គឺលេខសូន្យ (0) ហើយលក្ខណៈទីពីរតម្លៃខ្ពស់នៃ Digit មួយគឺតូចជាងតម្លៃរបស់គោលលេខ ឧទាហរណ៍ នៅក្នុងប្រព័ន្ធគោលដប់ គឺលេខ 9 ដែលមានតម្លៃតូចជាងប្រព័ន្ធគោលរបស់វាមួយលេខ។

៣.៣ ប្រព័ន្ធគោលពីរ (Binary Number System)

ប្រព័ន្ធគោលពីរ គឺជាប្រព័ន្ធគោលដែលមានទម្រង់ដូចគ្នាបេះបិទទៅនឹងប្រព័ន្ធដទៃទៀតដែរ ដោយគ្រាន់តែវាមានតម្លៃតែពីរប៉ុណ្ណោះ គឺ លេខ 0 និង លេខ 1 *ហើយប្រើលេខ 2 ជំនួសលេខ 10* ដោយសារតែនៅក្នុងប្រព័ន្ធនេះគឺគ្មានលេខ 2 នោះទេ ទើបសម្រេចយកលេខ 10 ឲ្យមានតម្លៃស្មើលេខ 2។

ឧទាហរណ៍៖ ខាងក្រោមប្រើសម្រាប់គណនាតម្លៃលេខដែលបាននិយាយខាងលើថា 10 ក្នុងប្រព័ន្ធគោលពីរមានតម្លៃស្មើ 2 ក្នុងប្រព័ន្ធគោល 10

$(10)_2 = (?)_{10}$

កំណត់ទីតាំងដូចដែលបានរៀបរាប់ក្នុងចំណុច ៣.២

0 នៅទីតាំងទី 0

1 នៅទីតាំងទី 1

➔ $(1*2^1) + (0*2^0) \Leftrightarrow (1*2) + (0*0) = 2+0 = \mathbf{2}$

ចំណាំ៖ គ្រប់ Computer ត្រូវបានរក្សាទុកលេខតួអក្សរ និងតួអក្សរពិសេសជាទម្រង់គោលពីរ។ នៅពេលដែល Computer បានដឹងពីចំនួនជួរ Data ដែលមាននៅក្នុង Memory របស់ម៉ាស៊ីន។

តារាងរវាងប្រព័ន្ធគោល៨ ដែលមានតម្លៃក្នុងប្រព័ន្ធគោល ១០

Binary	Decimal
000	0
001	1
010	2
011	3
100	4
101	5
110	6
111	7

៣.៤ ប្រព័ន្ធគោលប្រាំបី (Octal Number System)

នៅក្នុងប្រព័ន្ធគោលប្រាំបី មាននិមិត្តសញ្ញាប្រាំបីតួ ឬប្រាំបីលេខ ដែលតាងដោយ 0, 1, 2, 3, 4, 5, 6, 7 ។ នៅក្នុងប្រព័ន្ធគោលប្រាំបី គឺយកតម្លៃទីតាំងមកកំណត់ជាស្វ័យគុណនៅលើគោលប្រាំបីនោះ។

ដូចគ្នានឹងប្រព័ន្ធគោលពីរផងដែរថា នៅក្នុងប្រព័ន្ធគោលប្រាំបី គឺមិនមានលេខ 8 ឬលេខ 9 នោះទេ គឺមានត្រឹមតែលេខ 7។

ដូច្នេះ តើលេខ ៨ នៅក្នុងប្រព័ន្ធគោលប្រាំបី គេកំណត់យកតម្លៃលេខប៉ុន្មាន ទើបស្មើនឹងតម្លៃលេខក្នុងប្រព័ន្ធគោល 10? ប្រាកដណាស់ថា គេនឹងយកលេខ 10 មកជំនួសលេខ 8។

យើងនឹងធ្វើការបកស្រាយ ហេតុដូចម្តេចបានជាលេខ 10 ក្នុងគោលប្រព័ន្ធគោលប្រាំបី មានតម្លៃស្មើនឹងតម្លៃលេខ 8 ក្នុងប្រព័ន្ធគោលដប់។

ឧទាហរណ៍៖ ខាងក្រោមប្រើសម្រាប់គណនាតម្លៃលេខដែលបាននិយាយខាងលើថា 10 ក្នុងប្រព័ន្ធគោលប្រាំបី មានតម្លៃស្មើ 10 ក្នុងប្រព័ន្ធគោល 10

$(10)_8=(\ ?\ )_{10}$

កំណត់ទីតាំងដូចដែលបានរៀបរាប់ក្នុងចំណុច ៣.២

0 នៅទីតាំងទី 0

1 នៅទីតាំងទី 1

$(1*8^1) + (0*8^0) \Leftrightarrow (1*8) + (0*0) = 8+0 = 8$

៣.៥ ប្រព័ន្ធគោលដប់ប្រាំមួយ (Hexadecimal Number System)

នៅក្នុងប្រព័ន្ធគោល 16 និមិត្តសញ្ញាដែលគេយកមកប្រើគឺ 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, A, B, C, D, E, F។ តម្លៃលេខពី 0 ដល់ 9 ជាប្រព័ន្ធគោល 10 ហើយ A រហូតដល់ F តាងឱ្យលេខពី 11 ដល់ 15 (A=10, B=11,C=12, D=13, E=14, F=15)។ ទីតាំងនីមួយៗនៃតម្លៃក្នុងប្រព័ន្ធនេះគេកំណត់ដោយស្វ័យគុណនៅលើគោល (16)។

នៅក្នុងប្រព័ន្ធគោលដប់ប្រាំមួយ គែបានជំនួសនូវលេខ 10 ជាអក្សរ A ពិតមែន ប៉ុន្តែចាំបាច់អ្នកត្រូវចងចាំថា នៅក្នុងប្រព័ន្ធគោលដប់ប្រាំមួយនេះ កាលណាបើលេខលើពី ៩ រហូតដល់ ១៥ គឺគេយកតួអក្សរ A រហូតដល់ F មកជំនួស។ ចុះបើលើសពីតម្លៃ ១៥ ឬ F វានឹងទៅជាយ៉ាងណាវិញ (ក្នុងករណីដែលយើងប្រើលេខរាល់ថ្ងៃគឺ ប្រព័ន្ធគោលដប់)។

៣.៦ របៀបបំលែងប្រព័ន្ធគោលនីមួយៗទៅជាប្រព័ន្ធគោលផ្សេងៗ

ប្រព័ន្ធគោល 10 វាមានសារៈសំខាន់ជាងគេនៅក្នុងប្រព័ន្ធគោលទាំងអស់នៅក្នុងប្រព័ន្ធ ដោយសារយើងប្រើប្រាស់នៅក្នុងជីវភាពប្រចាំថ្ងៃគែប្រើប្រាស់ប្រព័ន្ធគោល 10។ យើងមានវិធីសាស្ត្រជាច្រើនដើម្បីបំលែងពីប្រព័ន្ធគោលមួយទៅគោលមួយទៀត

- **វិធីដែលយកមកប្រើ**
  - បំលែងប្រព័ន្ធគោល 10 ទៅជាប្រព័ន្ធគោលផ្សេងទៀត
  - បំលែងប្រព័ន្ធគោលផ្សេងៗឱ្យទៅជាប្រព័ន្ធគោល 10

៣.៦.១ វិធីបំលែងពីប្រព័ន្ធគោលពីរទៅជាប្រព័ន្ធគោលដប់

- កំណត់តម្លៃលេខ
- កំណត់តម្លៃទីតាំង
- យកតម្លៃលេខនៃប្រព័ន្ធគោលនោះ មកលើកជាស្វ័យគុណនៃទីតាំងដែលវាស្ថិតនៅ
- យកតម្លៃលេខដែលវាស្ថិតនៅមកគុណនឹងតម្លៃដែលបានលើកជាស្វ័យគុណខាងលើ
- បូកតម្លៃលេខទាំងអស់ចូលគ្នានៅក្នុងប្រព័ន្ធគោល ២

តារាងតម្លៃតាមទីតាំងដែលត្រូវបានកំណត់នៅក្នុងប្រព័ន្ធគោលពីរ៖

ទីតាំង	**តម្លៃជួរ**
0	2^{0}=1
1	2^{1}=2
2	2^{2}=4
3	2^{3}=8
4	2^{4}=16
5	2^{5}=32
6	2^{6}=64
7	2^{7}=128
8	2^{8}=256
9	2^{9}=512
10	2^{10}=1014
11	2^{11}=2048
12	2^{12}=4096

ឧទាហរណ៍៖ ចូរបំលែង $(11001)_2 = (\ ?\ )_{10}$

1) កំណត់ជួរឈរ

$2^4 = 16, 2^3=8, 2^2=4, 2^1=2, 2^0=1$

2) គុណតម្លៃលេខ

$1*2^4, 1*2^3, 0*2^2, 0*2^1, 1*2^0$

3) បូកតម្លៃចូលគ្នា

16+8+0+0+1 = 25

➔ $(11001)_2= (25)_{10}$

ក្រោយពីបានអនុវត្តន៍ឧទាហរណ៍ខាងលើរួចមក អ្នកទាំងអស់គ្នាអាចនឹងគិតថាតម្លៃលេខនៅក្នុងប្រព័ន្ធគោលពីរណាដែលមានតម្លៃសូន្យ (0) នៅពេលដែលយកទៅគុណនឹងតម្លៃដែលបានលើកជាស្វ័យគុណរួចនោះ វានឹងបង្ហាញតម្លៃសូន្យផងដែរ។

ហេតុនេះ យើងអាចកំណត់ម្យ៉ាងទៀតបានថា គឺយើងនឹងយកតែតម្លៃលេខណាដែលមិនមែនជាតម្លៃលេខសូន្យ (0) មកលើកជាស្វ័យគុណ បន្ទាប់មកបូកបញ្ចូលគ្នាដើម្បីទទួលបានលទ្ធផល ជាការស្រេច។

ឧទាហរណ៍ $(101)_2 = (\ ?\ )_{10}$

➔ $2^0 = 1, 2^2 = 4 => 1 + 4 = 5$

➔ ដូចនេះ $(101)_2 = (5)_{10}$

៣.៦.២ វិធីបំលែងពីប្រព័ន្ធគោលប្រាំបីទៅជាប្រព័ន្ធគោលដប់

- កំណត់តម្លៃលេខ
- កំណត់តម្លៃទីតាំង
- យកតម្លៃលេខនៃប្រព័ន្ធគោលនោះ មកលើកជាស្វ័យគុណនៃទីតាំងដែលវាស្ថិតនៅ
- យកតម្លៃលេខដែលវាស្ថិតនៅមកគុណនឹងតម្លៃដែលបានលើកជាស្វ័យគុណខាងលើ
- បូកតម្លៃលេខទាំងអស់ចូលគ្នានៅក្នុងប្រព័ន្ធគោលនោះ

តារាងតម្លៃតាមទីតាំងដែលត្រូវបានកំណត់នៅក្នុងប្រព័ន្ធគោលប្រាំបី៖

ទីតាំង	**តម្លៃជួរ**
0	$8^0=1$
1	$8^1=8$
2	$8^2=64$
3	$8^3=512$
4	$8^4=4096$
5	$8^5=32768$

ចូរបំលែង $(4706)_8 = (\ ?\ )_{10}$

1- $(4*8^3) + (7*8^2) + (0*8^1) + (6*8^0)$

2- (4*512) + (7*64) + (0*8) + 6*1)

3- 2048 + 448 + 0 + 6 = 2502

=> $(4706)_8 = (2502)_{10}$

៣.៦.៣ វិធីបំលែងពីប្រព័ន្ធគោលដប់ប្រាំមួយទៅជាប្រព័ន្ធគោលដប់

- កំណត់តម្លៃលេខ
- កំណត់តម្លៃទីតាំង
- យកតម្លៃលេខនៃប្រព័ន្ធគោលនោះ មកលើកជាស្វ័យគុណនៃទីតាំងដែលវាស្ថិតនៅ
- យកតម្លៃលេខដែលវាស្ថិតនៅមកគុណនឹងតម្លៃដែលបានលើកជាស្វ័យគុណខាងលើ
- បូកតម្លៃលេខទាំងអស់ចូលគ្នានៅក្នុងប្រព័ន្ធគោលនោះ

តារាងតម្លៃតាមទីតាំងដែលត្រូវបានកំណត់នៅក្នុងប្រព័ន្ធគោលប្រាំបី៖

ទីតាំង	តម្លៃជួរ
0	16^0=1
1	16^1=8
2	16^2=256
3	16^3=4096
4	16^4=65536
5	16^5=1048576
6	16^6=16777216
7	16^7=268435456
8	16^8=4294967296
9	16^9=68719476736

ចូរបំលែង $(1BF)_{16} = (\ ?\ )_{10}$

$(1*16^2) + (11*16^1) + (15*16^0)$

$(1*256) + (11*16) + (15*1)$

256 + 176 + 15 = 447

– $(1BF)_{16} = (447)_{10}$

៣.៧ របៀបបំលែងប្រព័ន្ធគោលដប់ ទៅជាប្រព័ន្ធគោលថ្មីទៀត

យើងមាន ៤ ដំណាក់កាលដើម្បីបំលែងពីគោល 10 ទៅជាគោលថ្មីទៀត៖

1- យើងចែកតម្លៃលេខនៅក្នុងគោល 10 ទៅចែកជាមួយគោលថ្មីដែលយើងចង់បំលែងទៅ

2- កត់ត្រា និងរក្សាទុកនូវសំណល់ដែលបានចែកសល់នោះទុក ដោយមិនធ្វើការបន្ថែមសូន្យអ្វីនោះទេ

3- ចែកតម្លៃលេខប្រព័ន្ធគោល ១០ នោះរហូតទាល់តែលទ្ធផលស្មើសូន្យ ទើបបញ្ចប់ការចែក

4- រៀបតម្លៃលេខដែលជាសំណល់ពីក្រោមឡើងមកលើវិញ ដើម្បីទទួលបានជាលទ្ធផល

<u>ចំណាំ៖</u>

1. វិធីបំលែងពីប្រព័ន្ធផ្សេងៗមកប្រព័ន្ធគោលដប់ គឺត្រូវប្រើប្រមាណវិធីគុណ

2. វិធីបំលែងពីប្រព័ន្ធគោលដប់ទៅប្រព័ន្ធគោលដទៃ គឺត្រូវប្រើប្រមាណវិធីចែក

ឧទាហរណ៍៖ ចូរបំលែងតម្លៃលេខនៅក្នុងប្រព័ន្ធគោលដប់ ទៅជាប្រព័ន្ធគោលពីរ

$(99)_{10} = (\ ? \ )_2$

យកតម្លៃលេខក្នុងប្រព័ន្ធគោលដប់នោះមកធ្វើការចែកនឹងប្រព័ន្ធដែលចង់បំលែងទៅ

1- 99 / 2 = 49 សំណល់ 1

2- 49 / 2 = 24 សំណល់ 1

3- 24 / 2 = 12 សំណល់ 0

4- 12 / 2 = 6 សំណល់ 0

5- 6 / 2 = 3 សំណល់ 0

6- 3 / 2 = 1 សំណល់ 1

7- 1 / 2 = 0 សំណល់ 1

រៀបតម្លៃលេខនៃសំណល់បញ្ច្រាស់ពីក្រោមទៅលើ ធ្វើជាលទ្ធផល

– **$(99)_{10} = (1100011)_2$**

ឧទាហរណ៍៖ ចូរបំលែងតម្លៃលេខនៅក្នុងប្រព័ន្ធគោលដប់ ទៅជាប្រព័ន្ធគោលប្រាំបី

$(99)_{10} = (\ ? \ )_8$

1. 99 / 8 = 12 សំណល់ 3
2. 12 / 8 = 1 សំណល់ 4
3. 1 / 8 = = សំណល់ 1

រៀបតម្លៃលេខនៃសំណល់បញ្ច្រាស់ពីក្រោមទៅលើ ធ្វើជាលទ្ធផល

→ $(99)_{10} = (143)_8$

ឧទាហរណ៍៖ ចូរបំលែងតម្លៃលេខនៅក្នុងប្រព័ន្ធគោលដប់ ទៅជាប្រព័ន្ធគោលដប់ប្រាំមួយ

$(99)_{10} = (\ ?\ )_{16}$

1. 99 / 16 = 6 សំណល់ 3
2. 6 / 16 = 0 សំណល់ 6

រៀបតម្លៃលេខនៃសំណល់បញ្ច្រាស់ពីក្រោមទៅលើ ធ្វើជាលទ្ធផល

→ $(99)_{10} = (63)_{16}$

៣.៨ របៀបបំលែងប្រព័ន្ធគោលពីរ ទៅជាប្រព័ន្ធគោលប្រាំបី

យើងបានដឹងហើយថាតម្លៃលេខក្នុងប្រព័ន្ធគោលប្រាំបី គឺមាន ៨ លេខ ដែលយើងយកមកប្រើប្រាស់ ហើយវាមានតម្លៃស្រដៀងនឹងប្រព័ន្ធគោលដប់ដែរ ដូចនេះការចងចាំតារាងជាការប្រសើរ ព្រោះតម្លៃលេខដែលគេយកប្រើ គឺពី 0 ដល់ ៧ ហើយបំបែកទៅជាប្រព័ន្ធគោលពីរ គឺមានបីតួ។

ដើម្បីបំលែងតម្លៃលេខក្នុងប្រព័ន្ធគោលពីរ ទៅជាប្រព័ន្ធគោលប្រាំបី គឺអ្នកអាចអនុវត្តតាមជំហាននៃការបំបែកបែបងាយៗដូចខាងក្រោម៖

- បែងចែកបណ្តុំតួរបស់ប្រព័ន្ធគោលពីរ ជាក្រុមៗដែលមានបីតួ (3 bits) រាប់ពីស្តាំ
  - ក្នុងករណីចំនួនតួដែលនៅសល់ គឺមិនគ្រប់ បីតួ អ្នកចាំបាច់បន្ថែម 0 ពីមុខតម្លៃលេខនោះឲ្យគ្រប់បីតួ (ឧទាហរណ៍៖ នៅសល់លេខ 1 ចាំបាច់ត្រូវបន្ថែមដោយស្វ័យប្រវត្តិពីខាងមុខលុខ 1 នោះដោយ 0 → 001 ព្រោះលេខសូន្យនៅពីមុខ គឺគ្មានន័យនោះទេ គ្រាន់តែយើងបន្ថែមវាឲ្យគ្រប់ទៅតាមលក្ខខណ្ឌរបស់គេ)
- បំលែងប្រព័ន្ធគោលពីរ នៃក្រុមនីមួយៗទៅជាចំនួន Octal digit។

ឧទាហរណ៍៖ $(101001)_2 = (\ ?\ )_8$

001 = 2

101 = 6

– $(101001)_2 = (62)_8$

ឧទាហរណ៍៖ $(1010011)_2 = (\ ?\ )_8$

011 = 3

010 = 2

លេខខាងមុខនៅសល់តែមួយខ្ទង់ គឺលេខ 1 កាលណាតម្លៃលេខដែលយើងបានកាត់ពីខាងស្តាំមកឆ្វេងវិញ ហើយនៅសល់តួមិនគ្រប់បីខ្ទង់ នោះអ្នកទាំងអស់គ្នាចាំបាច់ត្រូវបន្ថែមលេខសូន្យពីខាងមុខតម្លៃលេខនោះឲ្យគ្រប់ចំនួនតួរបស់វា គឺបីខ្ទង់ ដែល 1 → 001 = 1

=> $(1010011)_2 = (123)_8$

៣.៩ របៀបបំលែងប្រព័ន្ធគោលប្រាំបី ទៅជាប្រព័ន្ធគោលពីរ

ចំពោះតម្លៃលេខនៅក្នុងប្រព័ន្ធគោលពីរ គឺមានតែលេខ 0 និង 1 ប៉ុណ្ណោះ ដូចនេះដើម្បីបំលែងទៅប្រព័ន្ធគោលដប់ ទៅជាគោលពីរវិញ ចាំបាច់ត្រូវយកតម្លៃលេខដែលចង់បំលែងនោះ មកចែកនឹង ២ ដែលជាប្រព័ន្ធគោលនោះ បន្ទាប់មករក្សាទុកសំណល់ រាប់សំណល់ពីក្រោមឡើងមកខាងលើវិញធ្វើជាលទ្ធផលចុងក្រោយ។

ដោយឡែក ដើម្បីបំលែងតម្លៃលេខប្រព័ន្ធគោលប្រាំបី ទៅជាប្រព័ន្ធគោលពីរ គឺមានភាពងាយស្រួលជាងការបំលែងពីប្រព័ន្ធគោលដទៃ ដោយអ្នកគ្រាន់តែយកតម្លៃលេខនៅក្នុងប្រព័ន្ធគោលប្រាំបី ដែលចង់បំលែងនោះ ម្តងមួយលេខ ៗ មកបំបែកដោយការចាំតារាង ទៅជាប្រព័ន្ធគោលពីរ គឺបីខ្ទង់ជាការស្រេច។

ដើម្បីបំលែងតម្លៃលេខក្នុងប្រព័ន្ធគោលពីរ ទៅជាប្រព័ន្ធគោលប្រាំបី អ្នកត្រូវអនុវត្តតាមជំហានងាយៗដូចខាងក្រោម៖

- យកតម្លៃលេខនីមួយៗនៃប្រព័ន្ធគោលប្រាំបីនោះមកបំបែកជាប្រព័ន្ធគោលពីរ តាមការចងចាំតារាង
- គ្រប់លទ្ធផលដែលបានមកពីការបំលែងខាងលើ អ្នកត្រូវយកមករៀបធ្វើជាលទ្ធផល

ឧទាហរណ៍៖ ចូរបំលែង $(570)_8=(\ ?\ )_2$

5=101

7=111

0=000

=> $(570)_8=(101111000)_2$

ឧទាហរណ៍៖ ចូរបំលែង $(1452)_8=(\ ?\ )_2$

1 = 001

4 = 100

5 = 101

2 = 010

=> $(1452)_8 = (001100101010)_2 \rightarrow (1100101010)_2$

៣.១០ របៀបបំលែងប្រព័ន្ធគោលដប់ប្រាំមួយ ទៅជាប្រព័ន្ធគោលពីរ

ចំពោះតម្លៃលេខនៅក្នុងប្រព័ន្ធគោលពីរ គឺមានតែលេខ 0 និងលេខ 1 ប៉ុណ្ណោះ ដូចនេះដើម្បីបំលែងទៅប្រព័ន្ធគោលដប់ ទៅជាគោលពីរវិញ ចាំបាច់ត្រូវយកតម្លៃលេខដែលចង់បំលែងនោះ មកចែកនឹង ២ ដែលជាប្រព័ន្ធគោលនោះ បន្ទាប់មករក្សាទុកសំណល់ ដោយមិនត្រូវបន្ថែមសូន្យដើម្បីចែកបន្តនោះទេ គឺអាចរក្សាទុកសំណល់នោះ រួចធ្វើការរៀបនូវសំណល់នោះ ពីក្រោមឡើងមកខាងលើវិញធ្វើជាលទ្ធផលចុងក្រោយ។

ដោយឡែក ដើម្បីបំលែងតម្លៃលេខប្រព័ន្ធគោលដប់ប្រាំមួយ ទៅជាប្រព័ន្ធគោលពីរ គឺមានភាពងាយស្រួលជាងការបំលែងពីប្រព័ន្ធគោលដប់ មកជាប្រព័ន្ធគោលពីរ ដោយសារអ្នកអាចយកតម្លៃលេខនៅក្នុងប្រព័ន្ធគោលដប់ប្រាំមួយនីមួយៗដែលចង់បំលែងនោះ ម្តងមួយលេខៗ មកបំបែកដោយការចងចាំនូវតម្លៃលេខក្នុងតារាងទៅជាប្រព័ន្ធគោលពីរ គឺបួនខ្ទង់ ជាការស្រេច។

ដើម្បីបំលែងតម្លៃលេខក្នុងប្រព័ន្ធគោលពីរ ទៅជាប្រព័ន្ធគោលដប់ប្រាំមួយ អ្នកត្រូវអនុវត្តតាមជំហានងាយៗដូចខាងក្រោម៖

- យកតម្លៃលេខដែលមាននៅក្នុងប្រព័ន្ធគោលដប់ប្រាំមួយ នីមួយៗនោះ មកបំបែកទៅជាប្រព័ន្ធគោលពីរ
- តំរៀបលេខដែលបានបំបែកខាងលើនោះធ្វើជាលទ្ធផលតែម្តង

ឧទាហរណ៍៖: ចូរបំលែង $(8AF)_{16}=(\ ?\ )_2$

8 = 1000

A (10) = 1010

F (15) = 1111

=> $(8AF)_{16}= (100010101111)_2$

៣.១១ របៀបបំលែងប្រព័ន្ធគោលពីរ ទៅជាប្រព័ន្ធគោលដប់ប្រាំមួយ

តម្លៃលេខដែលមាននៅក្នុងប្រព័ន្ធគោលពីរ គឺមានតែ 0 និង ១ ប៉ុណ្ណោះ ហើយសម្រាប់ការបំបែកទៅប្រព័ន្ធផ្សេងទៀត ដូចជាប្រព័ន្ធគោលដប់ គឺចាំបាច់ណាស់ ដែលអ្នកត្រូវយកលេខនៃប្ញព័ន្ធគោលពីរនោះ មកធ្វើការរៀបចំដាក់ស្វ័យគុណ និងបូកបញ្ចូលគ្នា។

ដោយឡែកសម្រាប់ការបំលែងតម្លៃលេខ ពីប្រព័ន្ធគោលពីរ ទៅជាប្រព័ន្ធគោលដប់ប្រាំមួយ គឺអាចនិយាយបានថាវាមានភាពងាយស្រួល និងលឿនជាងការបំបែកទៅជាប្រព័ន្ធគោលដប់។ ដោយសារថាអ្នកអាចយកតម្លៃលេខដែលមាននៅក្នុងប្រព័ន្ធគោលពីរទាំងអស់ដែលអ្នកចង់យកមកបំបែកនោះ មករៀបជាផ្នែកតូច គឺម្តង ៤ខ្ទង់ នៃប្រព័ន្ធគោលពីរនោះ ដែលជាលទ្ធផលនឹងទទួលបានយ៉ាងងាយ។

ដើម្បីបំលែងតម្លៃលេខក្នុងប្រព័ន្ធគោលពីរ ទៅជាប្រព័ន្ធគោលដប់ប្រាំមួយ អ្នកត្រូវអនុវត្តតាមជំហានងាយៗដូចខាងក្រោម៖

- រៀបតម្លៃលេខនៃប្រព័ន្ធគោលពីរនោះពីខាងស្តាំមក ម្តង៤ខ្ទង់ រួចបំបែកទៅជាប្រព័ន្ធគោលដប់ប្រាំមួយ តាមការចងចាំតារាង
- យកតម្លៃដែលបានបំបែកពីខាងលើ មកតំរៀបទៅតាមលំដាប់វិញធ្វើជាលទ្ធផល

ឧទាហរណ៍៖ $(1001101010101)_2 = (\ ?\ )_{16}$

0101 = 5

0101 = 5

0011 = 4

1 → 0001 = 1

=> $(1001101010101)_2 = (1455)_{16}$

៣.១២ របៀបបំលែងប្រព័ន្ធគោលប្រាំបី ទៅជាប្រព័ន្ធគោលដប់ប្រាំមួយ

វិធីសាស្ត្រដែលយើងបានសិក្សា និងអានពីខាងលើមក គឺមានច្រើនរបៀប និងច្រើនសណ្ឋានដើម្បីអនុវត្ត ដូចជាការបំលែងពីប្រព័ន្ធគោលដប់ ទៅប្រព័ន្ធផ្សេងទៀតដូចជា ប្រព័ន្ធគោលពីរ ប្រព័ន្ធគោលប្រាំបី និងប្រព័ន្ធគោលដប់ប្រាំមួយ។ ក៏មានផងដែរនូវការបំលែងត្រលប់មកវិញពីប្រព័ន្ធគោលទាំងនោះ មកជាប្រព័ន្ធគោលដប់ ដូចដែលបានបង្ហាញជារបៀបនៃការអនុវត្ត និងមានជាឧទាហរណ៍មកបង្ហាញផងដែរ។

ដោយឡែកការបំលែងពីប្រព័ន្ធគោលពីរ ទៅជាប្រព័ន្ធគោលប្រាំបី និងបំលែងពីប្រព័ន្ធគោលពីរ ទៅប្រព័ន្ធគោលដប់ប្រាំមួយ ព្រមទាំងការបំលែងត្រលប់មកវិញ គឺមានភាពងាយស្រួលដូចដែលបានបង្ហាញរួចស្រាប់ហើយ។

សម្រាប់ចំណុចនេះ យើងនឹងធ្វើការបកស្រាយអំពីការបំលែងតម្លៃលេខពីប្រព័ន្ធគោលប្រាំបី ទៅជាប្រព័ន្ធគោលដប់ប្រាំមួយវិញម្តង។

ដើម្បីអនុវត្តការងារបំលែងលេខនៃប្រព័ន្ធគោលប្រាំបី ទៅជាតម្លៃលេខប្រព័ន្ធគោលដប់ប្រាំមួយ អ្នកត្រូវអនុវត្តតាមជំហានដូចដែលបានរៀបរាប់ខាងក្រោម៖

1) យកតម្លៃលេខនីមួយៗនៃប្រព័ន្ធគោលប្រាំបី មកបំលែងទៅជាប្រព័ន្ធគោលពីរ
2) រៀបតម្លៃលេខនៃប្រព័ន្ធគោលពីរដែលបានបំលែងចេញពីប្រព័ន្ធគោលប្រាំបីខាងលើដាក់បញ្ចូលគ្នា
3) បែងចែកតម្លៃលេខនៃប្រព័ន្ធគោលពីរដែលបានរៀបរួចទៅជា បួនតួ រួចមើលតារាងដើម្បីបញ្ចេញលទ្ធផលនោះជាការស្រេច

ឧទាហរណ៍៖ ចូរបំលែងតម្លៃលេខប្រព័ន្ធគោលប្រាំបី ទៅជាតម្លៃលេខប្រព័ន្ធគោលដប់ប្រាំមួយ

$(567)_8 = (\ ?\ )_{16}$

បំលែង $(567)_8$ ទៅជាប្រព័ន្ធគោលពីរជាមុនសិន

5 = 101

6 = 110

7 = 111

→ $(567)_8 = (101110111)_2$

បំលែង $(101110111)_2$ ទៅជាប្រព័ន្ធគោលដប់ប្រាំមួយវិញម្តង

0111 = 7

0111 = 7

0001 = 1

→ $(567)_8 = (177)_{16}$

៣.១៣ របៀបបំលែងប្រព័ន្ធគោលដប់ប្រាំមួយ ទៅជាប្រព័ន្ធគោលប្រាំបី

វិធីសាស្ត្រដែលយើងបានសិក្សា និងអានពីខាងលើមក គឺមានច្រើនរបៀប និងច្រើនសណ្ឋានដើម្បីអនុវត្ត ដូចជាការបំលែងពីប្រព័ន្ធគោលដប់ ទៅប្រព័ន្ធផ្សេងទៀតដូចជា ប្រព័ន្ធគោលពីរ ប្រព័ន្ធគោលប្រាំបី និងប្រព័ន្ធគោលដប់ប្រាំមួយ។ ក៏មានផងដែរនូវការបំលែងត្រលប់មកវិញពីប្រព័ន្ធគោលទាំងនោះ មកជាប្រព័ន្ធគោលដប់ ដូចដែលបានបង្ហាញជារបៀបនៃការអនុវត្ត និងមានជាឧទាហរណ៍មកបង្ហាញផងដែរ។

ដោយឡែកការបំលែងពីប្រព័ន្ធគោលពីរ ទៅជាប្រព័ន្ធគោលប្រាំបី និងបំលែងពីប្រព័ន្ធគោលពីរ ទៅប្រព័ន្ធគោលដប់ប្រាំមួយ ព្រមទាំងការបំលែងត្រលប់មកវិញ គឺមានភាពងាយស្រួលដូចដែលបានបង្ហាញរួចស្រាប់ហើយ។

សម្រាប់ចំណុចនេះ យើងនឹងធ្វើការបកស្រាយអំពីការបំលែងតម្លៃលេខពីប្រព័ន្ធគោលដប់ប្រាំមួយទៅជាប្រព័ន្ធគោលប្រាំបី វិញម្តង។

ដើម្បីអនុវត្តការងារបំលែងលេខនៃប្រព័ន្ធគោលដប់ប្រាំមួយ ទៅជាតម្លៃលេខប្រព័ន្ធគោលប្រាំបី អ្នកត្រូវអនុវត្តតាមជំហានដូចដែលបានរៀបរាប់ខាងក្រោម៖

1) យកតម្លៃលេខនីមួយៗនៃប្រព័ន្ធគោលគោលដប់ប្រាំមួយ មកបំលែងទៅជាប្រព័ន្ធគោលពីរ
2) រៀបតម្លៃលេខនៃប្រព័ន្ធគោលពីរ ដែលបានបំលែងចេញពីប្រព័ន្ធគោលដប់ប្រាំមួយខាងលើដាក់បញ្ចូលគ្នា
3) បែងចែកតម្លៃលេខនៃប្រព័ន្ធគោលពីរដែលបានរៀបរួចទៅជា បីតួ រួចមើលតារាងដើម្បីបញ្ចេញលទ្ធផលនោះជាការស្រេច

ឧទាហរណ៍៖ ចូរបំលែងតម្លៃលេខប្រព័ន្ធគោលប្រាំបី ទៅជាតម្លៃលេខប្រព័ន្ធគោលដប់ប្រាំមួយ

$(567)_8 = (\ ?\ )_{16}$

បំលែង $(A9B8)_{16}$ ទៅជាប្រព័ន្ធគោលពីរជាមុនសិន

A = 1010

9 = 1001

B = 1011

8 = 1000

→ $(A9B8)_{16} = (1010100110111000)_2$

បំលែង $(1010100110111000)_2$ ទៅជាប្រព័ន្ធគោលប្រាំបីវិញម្តង

000 = 0

111 = 7

110 = 6

100 = 4

010 = 2

1 → 001 = 1

→ $(A9B8)_{16} = (124670)_8$

៣.១៤ ចំនួនទសភាគ (fraction number)[11]

យើងបានដឹងរួចមកហើយថា ទំនាក់ទំនងមូលដ្ឋានរួម ប្រកាន់យកសម្រាប់គោលមួយចំនួន ហើយការពង្រីកទៅលើ Digits នីមួយៗគឺ 10 ដង នៃការពង្រីកទៅលើផ្នែកខាងស្តាំ។

ដូចដែលយើងបានពិភាក្សាខាងលើ គឺថាយើងត្រូវគុណ ឬចែកនឹងលេខប្រភាគទៅនឹងប្រព័ន្ធលេខណាដែលយើងចង់បំលែង។

ដូច្នេះនៅទីនេះយើងបម្លែងលេខទសភាគនៅក្នុងប្រព័ន្ធគោលពីរ គឺចាំបាច់យើងត្រូវយកតម្លៃលេខដែលនៅខាងក្រោយក្បៀសនោះ មកគុណនឹង ២ រហូតដល់តម្លៃក្រោយក្បៀសនោះស្មើសូន្យ ឬគុណចំនួនបីដង ទើបយើងបញ្ចប់ការគុណនោះ។

បម្លែងលេខទសភាគនៅក្នុងប្រព័ន្ធគោលប្រាំបី គឺចាំបាច់យើងត្រូវយកតម្លៃលេខដែលនៅខាងក្រោយក្បៀសនោះ មកគុណនឹង ៨។

បម្លែងលេខទសភាគនៅក្នុងប្រព័ន្ធគោលដប់ប្រាំមួយ គឺចាំបាច់យើងត្រូវយកតម្លៃលេខដែលនៅខាងក្រោយក្បៀសនោះ មកគុណនឹង ១៦។

[11] https://www.electronicsengineering.nbcafe.in/step-step-process-fractional-number-conversion-number-systems/

ប្រព័ន្ធគោលពីរ លេខទសភាគ

2^0 =	$\frac{1}{2^0}$	$\frac{1}{1}$	1
2^{-1} =	$\frac{1}{2^1}$	$\frac{1}{2}$	0.5
2^{-2} =	$\frac{1}{2^2}$	$\frac{1}{4}$	0.25
2^{-3} =	$\frac{1}{2^3}$	$\frac{1}{8}$	0.125
2^{-4} =	$\frac{1}{2^4}$	$\frac{1}{16}$	0.0625
2^{-5} =	$\frac{1}{2^5}$	$\frac{1}{32}$	0.03125

៣.១៤.១ បំលែងតម្លៃទសភាគ ពីប្រព័ន្ធគោលដប់ ទៅជាប្រព័ន្ធគោលពីរ

ដូចដែលយើងបានរៀបរាប់ពីខាងលើថាយើងត្រូវយកតម្លៃលេខដែលយើងចង់បំបែកនោះទៅលើកជាស្វ័យគុណ រួចបូកបញ្ចូលគ្នា ឬចែកយកតម្លៃលេខទាំងនោះមកចែក និងរៀបសំណល់បញ្ច្រាស ដើម្បីទទួលជាលទ្ធផល។

ដោយឡែកនៅក្នុងការបំលែងតម្លៃលេខទសភាគនេះវិញ យើងនឹងបម្លែងតម្លៃលេខប្រភាគនោះទៅតាមការគុណនឹង ២ ដែលជាលេខគោលនៃប្រព័ន្ធ។

ដើម្បីបំលែងតម្លៃលេខទសភាគ ឬហៅថាតម្លៃក្រោយក្បៀស អ្នកត្រូវអនុវត្តតាមជំហានដូចខាងក្រោម៖

1) យកតែដែលនៅក្រោយក្បៀសចេញមកគណនា
2) យកតម្លៃដែលបានយកចេញពីខាងលើនោះមកគុណនឹងលេខគោលនៃប្រព័ន្ធគោលពីរនោះ
3) គុណរហូតក្រោយក្បៀសស្មើសូន្យ ឬគុណចំនួនបីដង

ឧទាហរណ៍៖ បំលែងតម្លៃលេខប្រព័ន្ធគោល ១០ ទៅជាប្រព័ន្ធគោល ២

$(0.25)_{10} = (\ ?\ )_2$

- 0.25 * 2 = 0.50 → ***0***
- 0.50 * 2 = 1.00 → ***1***

- → $(0.25)_2 = (0.01)_2$

ឧទាហរណ៍៖ បំលែងតម្លៃលេខប្រព័ន្ធគោល ១០ ទៅជាប្រព័ន្ធគោល ២

$(0.5)_{10} = (?)_2$

- 0.5 * 2 = 1.00 → $(0.5)_{10} = (0.1)_2$

៣.១៤.២ បំលែងតម្លៃទសភាគ ពីប្រព័ន្ធគោល១០ ទៅជាប្រព័ន្ធគោល៨

ដោយឡែកនៅក្នុងការបំលែងតម្លៃលេខទសភាគនេះវិញ យើងនឹងបម្លែងតម្លៃលេខប្រភាគនោះទៅតាមការគុណនឹង ៨ ដែលជាលេខគោលនៃប្រព័ន្ធ។

ដើម្បីបំលែងតម្លៃលេខទសភាគ ឬហៅថាតម្លៃក្រោយក្បៀស អ្នកត្រូវអនុវត្តតាមជំហានដូចខាងក្រោម៖

1) យកតែដែលនៅក្រោយក្បៀសចេញមកគណនា
2) យកតម្លៃដែលបានយកចេញពីខាងលើនោះមកគុណនឹងលេខគោលនៃប្រព័ន្ធគោល ៨ នោះ
3) គុណរហូតក្រោយក្បៀសស្មើសូន្យ ឬគុណចំនួនបីដង

ឧទាហរណ៍៖ បំលែងតម្លៃលេខប្រព័ន្ធគោល ១០ ទៅជាប្រព័ន្ធគោល ៨

$(0.25)_{10} = (?)_8$

- 0.25 * 8 = 2
- → $(0.25)_{10} = (0.2)_8$

៣.១៤.៣ បំលែងតម្លៃទសភាគ ពីប្រព័ន្ធគោល១០ ទៅជាប្រព័ន្ធគោល១៦

ដោយឡែកនៅក្នុងការបំលែងតម្លៃលេខទសភាគនេះវិញ យើងនឹងបម្លែងតម្លៃលេខប្រភាគនោះទៅតាមការគុណនឹង ១៦ ដែលជាលេខគោលនៃប្រព័ន្ធ។

ដើម្បីបំលែងតម្លៃលេខទសភាគ ឬហៅថាតម្លៃក្រោយក្បៀស អ្នកត្រូវអនុវត្តតាមជំហានដូចខាងក្រោម៖

1) យកតែដែលនៅក្រោយក្បៀសចេញមកគណនា
2) យកតម្លៃដែលបានយកចេញពីខាងលើនោះមកគុណនឹងលេខគោលនៃប្រព័ន្ធគោល ១៦ នោះ
3) គុណរហូតក្រោយក្បៀសស្មើសូន្យ ឬគុណចំនួនបីដង

ឧទាហរណ៍៖ បំលែងតម្លៃលេខប្រព័ន្ធគោល ១០ ទៅជាប្រព័ន្ធគោល ១៦

$(0.25)_{10} = (?)_{16}$

- 0.25 * 16 = 4
- → $(0.25)_{10} = (0.4)_{16}$

តារាងបង្ហាញពីទំនាក់ទំនងរវាងប្រព័ន្ធគោលទាំង ៤

Hexadecimal	Decimal	Octal	Binary
0	0	0	0
1	1	1	1
2	2	2	10
3	3	3	11
4	4	4	100
5	5	5	101
6	6	6	110
7	7	7	111
8	8	10	1000
9	9		1001
A	10		1010
B	11		1011
C	12		1100
D	13		1101
E	14		1110
F	15		1111
10	16		1000

លំហាត់

1. ចូរគណនាតម្លៃលេខខាងក្រោម៖
   a. $(210)_{10} = (?)_8$
   b. $(1010001)_2 = (?)_{10}$
   c. $(72)_8 = (?)_2$
   d. $(0.256)_8 = (?)_{10}$
   e. $(11110011)_2 = (?)_8$
   f. $(1010111001)_2 = (?)_{16}$
   g. $(2576)_8 = (?)_{16}$
2. ចូរបំលែងលេខ binary $(1101101111110101)_2$ ទៅជាប្រព័ន្ធគោលដប់ប្រាំមួយ។
3. ចូរបំលែងលេខ ឧទាហរណ៍៖a decimal (ABD) ទៅជាប្រព័ន្ធគោល ១០ (Decimal)។
4. ចូរបំលែងលេខ ឧទាហរណ៍៖ ABC7 ទៅជា binary។
5. ចូរបំលែងលេខ Octal 752 ទៅជាប្រព័ន្ធគោល ២ (binary)។

មេរៀនទី ៤ ជំហាននៃការដោះស្រាយបញ្ហា

- តើអ្នកធ្លាប់ដែលបានស្គាល់នូវដំណោះស្រាយបញ្ហាណាមួយដែរឬទេ ?
- តើហេតុអ្វីបានជាយើងចាំបាច់ត្រូវសិក្សាពី Algorithm, Flowchart និង Pseudocode ?

រាល់ចំណោទបញ្ហានីមួយៗសុទ្ធតែមានដំណោះស្រាយជានិច្ច គ្រាន់តែយើងចាំបាច់ត្រូវធ្វើការឈ្វេងយល់ និងរកដំណោះស្រាយសម្រាប់បញ្ហា ឬរឿងនោះ។ អ្នកចាំបាច់នោះត្រូវយល់ឲ្យច្បាស់ និងរកដំណោះស្រាយសមស្របនឹងកាលៈទេសៈនោះផងដែរ។ នៅរាល់ដំណោះស្រាយនីមួយៗ តែងតែមានការចាប់ផ្ដើម និងការបញ្ចប់ទៅប្រកបដោយលក្ខខណ្ឌមួយដែលល្អ និងជោគជ័យ។

គោលបំណងនៃមេរៀន

នៅក្នុងមេរៀននេះ នឹងពន្យល់បកស្រាយនូវរាល់ការដោះស្រាយបញ្ហាពីជំហានងាយៗ រហូតដល់ជំហានដែលមានលក្ខណៈស៊ីជម្រៅក្នុងការរស់នៅ ក៏ដូចជាការងារជាក់ស្ដែងផងដែរ។

បន្ទាប់ពីអានមេរៀននេះរួច អ្នកនឹងទទួលបាន និងយល់នូវៈ

- កំណត់ថាតើក្បួនដោះស្រាយគឺជាអ្វី
- បំបែកបញ្ហាទៅជាលំដាប់នៃជំហានងាយៗ
- បង្ហាញជំហានទាំងនោះតាមក្រាហ្វិចដោយប្រើគំនូសតាងលំហូរ
- ពណ៌នាអំពីជំហានទាំងនោះជាពាក្យដោយប្រើជាភាសាអង់គ្លេស

៤.១ ក្បួនដោះស្រាយបញ្ហា (Algorithm)

ក្បួនដោះស្រាយបញ្ហា គឺជានីតិវិធីដែលបញ្ហាមួយ (ការគណនាឬផ្សេងទៀត) ត្រូវបានដោះស្រាយដោយកំណត់តាមច្បាប់ ឬជំហានជាក់លាក់។ នៅក្នុងផ្នែកវិទ្យាសាស្ត្រកុំព្យូទ័រជាច្រើនវគ្គសិក្សាទាំងមូល ដែលនឹងផ្តោតទៅលើការសិក្សាអំពីក្បួនដោះស្រាយបញ្ហា ដែលមានសារៈសំខាន់សម្រាប់ធ្វើការដោះស្រាយបញ្ហាជាច្រើនផងដែរនោះ។ ក្បួនដោះស្រាយមួយចំនួន មានសំខាន់ណាស់ រួមបញ្ចូលទាំងការដែលយើងនឹងត្រូវសិក្សាមួយផ្នែកពេញមួយឆមាស។

ក្បួនដោះស្រាយ គឺជាជំហានមួយសំណុំ ក្នុងការដោះស្រាយជាជំហាន ៗ សម្រាប់ធ្វើឲ្យសម្រេចនូវបញ្ហាដែលត្រូវដោះស្រាយ វាមានលក្ខណៈស្រដៀងនឹងភាសាសរសេរកម្មវិធីខ្លះដែរ ដែលគឺជាវិធីនៃការបង្ហាញក្បួនដោះស្រាយជាជំហានដំបូង។

ឧទាហរណ៍៖

- ការលុបបំបាត់នូវស្លាកស្នាមលើស្បែក
- តម្រៀបលេខ ពីធំ ទៅតូច និងពីតូច ទៅធំ
- ស្វែងរកតម្លៃតូចបំផុត ឬធំបំផុត
- រកចំនួនបឋម ឬចំនួនមិនបឋម
- តើតម្លៃលេខទាំងអស់នេះ តម្លៃលេខណាដែលជាលេខគូ និងតម្លៃលេខណាជាលេខសេស
- ការប្តូរប្រាក់ពីដុល្លារ ទៅជារៀល ឬពីរៀល ទៅជាដុល្លារ ។ល។

ក្បួនដោះស្រាយបញ្ហាមួយដែលកំពុងកើតមានឡើងក្នុងជីវភាពយើងរាល់ថ្ងៃគឺ៖

- *ចាប់ផ្តើមពិនិត្យមើលអាកាសធាតុ*
  - *ប្រសិនបើមានភ្លៀង*
    - *ជិះរថយន្តក្រុងទៅសាលារៀន*
  - *តែបើគ្មានភ្លៀងធ្លាក់នោះទេ*
    - *ដើរទៅសាលា ឬជិះកង់ទៅសាលា*
  - *បញ្ចប់ការពិនិត្យ និងការសម្រេចចិត្ត*
- *ចេញដំណើរ*

ក្បួនដោះស្រាយបញ្ហា គឺជាផែនការមួយក៏ពិតមែន ប៉ុន្តែជាសំណុំនៃការណែនាំ នេះក៏មានជាជំហានក្នុងការដោះស្រាយបញ្ហានីមួយៗ ដោយវាមានមូលដ្ឋានគ្រឹះចំនួនបី ក្នុងការអភិវឌ្ឍចលនារបស់ក្បួនដោះស្រាយរួមមាន៖

- ដំណោះស្រាយបញ្ហាទៅតាមលំដាប់លំដោយ (Sequence Algorithms)
- ដំណោះស្រាយបញ្ហាដោយមានលក្ខខណ្ឌដើម្បីផ្ទៀងផ្ទាត់ (Decision Algorithms)
- ដំណោះស្រាយបញ្ហាដោយឡើងវិញដដែលៗ ឬបន្តបន្ទាប់រហូតកន្លែងដែលត្រូវបញ្ចប់ (Repetition Algorithms)

៤.១.១ ដំណោះស្រាយបញ្ហាទៅតាមលំដាប់លំដោយ (Sequence Algorithms)

ការដោះស្រាយបញ្ហាទៅតាមលំដាប់លំដោយ គឺមានន័យជាការដោះស្រាយបញ្ហាបែបសាមញ្ញ ដែលធ្វើការងារពីរដើមទី រហូតដល់ចប់ដោយមិនបានគិតអំពីលក្ខខណ្ឌអ្វីនោះទេ។ ឧទាហរណ៍ ដូចជាការបូកលេខ គឺត្រូវមានចំនួនលេខពីរដើម្បីធ្វើការបូករកលទ្ធផល។

រូបមន្ត៖

1. Start
2. Input variables
3. Process statement
4. Output statement
5. Stop

មានន័យថាកាងារធ្វើពីលើមកក្រោមដោយមិនគិតពីលក្ខខណ្ឌអ្វី ឬផ្ទៀងផ្ទាត់អ្វីនោះទេ។ យើងអាចធ្វើការប្រៀបធៀបជាមួយនឹងទម្រង់នៃកូដជាមួយនឹងភាសា C++

1. #include<iostream>
2. using namespace std;
3. Int main(){
    a. Statements;
    b. Return 0;
4. }

ឧទាហរណ៍៖ ចូរសរសេរជំហាននៃការដោះស្រាយបញ្ហាដើម្បីគុណលេខពីរចំនួន

1. Start
2. Input a, b
3. Total = a * b
4. Output Total
5. Stop

យើងអាចធ្វើការប្រៀបធៀបជាមួយនឹងការសរសេរជាមួយនឹងភាសា C++

1- #include<iostream>

2- using namespace std;

3- int a, b, total;

4- int main(){

   a. total=a * b;

   b. cout<<"Result= "<<total<<endl;

5- return 0;

6- }

៤.១.២ ក្បួនដោះស្រាយបញ្ហាមានលក្ខខណ្ឌមួយ (If…Statement)

ដំណើរស្រាយបញ្ហាមានលក្ខខណ្ឌមួយ គឺប្រើប្រាស់ការងារសម្រាប់ផ្ទៀងផ្ទាត់លក្ខខណ្ឌមួយៗ ដើម្បីធ្វើការសន្និដ្ឋានថាតើវាពិត ឬមិនពិតដើម្បីបង្ហាញលទ្ធផលនោះ។

រូបមន្តៈ

1. Start
2. Input Variable
3. If(Expression) statement
4. Output statement
5. Stop

យើងអាចធ្វើការប្រៀបធៀបជាមួយនឹងទម្រង់នៃកូដជាមួយនឹងភាសា C++

1. #include<iostream>
2. using namespace std;
3. int variable;
4. Int main(){

a. If(Expression)

    i. Statements;

b. cout<<statements<<endl;

c. Return 0;

5. }

មានន័យថា៖

ប្រសិនបើលក្ខខណ្ឌ (Expression) នោះពិត វានឹងដំណើរការ Statements នោះ ប៉ុន្តែវានឹងមិនដំណើរការអ្វីនោះទេប្រសិនមិនបានជួបលក្ខខណ្ឌពិតនោះ។

ឧទាហរណ៍៖ ចូរសរសេរជំហាននៃការដោះស្រាយបញ្ហាដើម្បីគណនារកតម្លៃធំបំផុតនៃពីរចំនួន

1- Start

2- Input a, b

3- If (a > b) Max = a

4- If (a < b) Max = b

5- Output ("តម្លៃធំបំផុតនៃពីរចំនួន a និង b គឺ Max")

6- Stop

យើងអាចធ្វើការប្រៀបធៀបជាមួយនឹងការសរសេរជាមួយនឹងភាសា C++

1- #include<iostream>

2- using namespace std;

3- int a,b;

4- int main(){

5- if(a>b) cout<<"Maximum is a"<<endl;

6- if(a<b) cout<<"Maximum is b"<<endl;

7- return 0;

8- }

៤.១.៣ ក្បួនដោះស្រាយបញ្ហាមានលក្ខខណ្ឌពីរ (if…else statement)

ក្បួនដោះស្រាយបញ្ហាមានលក្ខខណ្ឌពីរ គឺត្រូវបានប្រើប្រាស់សម្រាប់ផ្ទៀងផ្ទាត់លក្ខខណ្ឌពីរ ដើម្បីធ្វើការងារនោះ ឬអាចនិយាយបានថា លក្ខខណ្ឌក្នុងដំណើរការនេះ គឺមានតែពីរប៉ុណ្ណោះ ឬអាចនិយាយបានថាជាការផ្ទៀងផ្ទាត់សម្រាប់លក្ខខណ្ឌ **ពិត ឬ មិនពិត** បើលក្ខខណ្ឌមួយ (if) មិនពិត នោះវានឹងអនុវត្តការងារដែលមាននៅក្នុងលក្ខខណ្ឌក្រៅពីលក្ខខណ្ឌមួយនោះ គឺវានឹងធ្វើការងារដែលមាននៅក្នុងលក្ខខណ្ឌផ្សេងពីនោះ (else) តែម្តង។

រូបមន្ត៖

1. Start
2. Input Variable
3. If(Expression)
    a. Process Statement1
4. Else
    a. Process Statement2
5. Output Statement
6. Stop

- យើងអាចធ្វើការប្រៀបធៀបជាមួយនឹងទម្រង់នៃកូដជាមួយនឹងភាសា C++

    1. #include<iostream>
    2. using namespace std;
    3. int variable;
    4. Int main(){
        a. If(Expression)
            i. Statements1;
        b. else

i. Statements2;

c. cout<<statements<<endl;

d. Return 0;

5. }

មានន័យថាៈ

ប្រសិនបើលក្ខខណ្ឌ (Expression) នោះពិត វានឹងដំណើរការអនុវត្ត Statement1 បើមិនដូច្នោះទេ គឺវានឹងអនុវត្ត Statement2 តែម្តង ព្រោះដូចយើងបានឃើញនៅក្នុង else គឺគ្មានលក្ខខណ្ឌនោះទេ។

ឧទាហរណ៍ៈ ចូរសរសេរជំហាននៃការដោះស្រាយបញ្ហា ដើម្បីគណនារកចំនួនធំបំផុតនៃពីរចំនួន

1- Start

2- Input a, b

3- If (a > b) Max = a

4- Else Max = b

5- Output ("តម្លៃធំបំផុតនៃពីរចំនួន a និង b គឺ Max")

6- Stop

- យើងអាចធ្វើការប្រៀបធៀបជាមួយនឹងការសរសេរជាមួយនឹងភាសា C++

1- #include<iostream>

2- using namespace std;

3- int a,b;

4- int main(){

5- if(a>b) cout<<"Maximum is a"<<endl;

6- else cout<<"Maximum is b"<<endl;

7- return 0;

8- }

៤.១.៤ ក្បួនដោះស្រាយបញ្ហាមានដែលមានលក្ខខណ្ឌច្រើនជាងពីរ (if… else-if…)

ដោះស្រាយបញ្ហាដែលមានលក្ខខណ្ឌច្រើនជាងពីរ គឺត្រូវបានគេយកមកប្រើប្រាស់សម្រាប់ការងារទាំងឡាយណាដែលមានលក្ខខណ្ឌច្រើនជាងពីរ ដើម្បីធ្វើការផ្ទៀងផ្ទាត់ បានន័យថាវានឹងផ្ទៀងផ្ទាត់ចាប់ពីលក្ខខណ្ឌទីមួយ លក្ខខណ្ឌទីពីរ លក្ខខណ្ឌទីបី ... រហូតដល់លក្ខខណ្ឌចុងក្រោយគឺ else ទើបវាបញ្ឈប់ការផ្ទៀងផ្ទាត់។

រូបមន្ត៖

1. Start
2. Input Variable
3. If (Expression1)
    a. Statement_If;
4. Else-if (Expression2)
    a. Statement_Else-if;
5. ……..
6. Else
    a. Statement_Else;
7. Output Statement
8. Stop

- យើងអាចធ្វើការប្រៀបធៀបជាមួយនឹងទម្រង់នៃកូដជាមួយនឹងភាសា C++

1. #include<iostream>
2. using namespace std;
3. int variable;
4. Int main(){
    a. If(Expression1)

        Statements1;

    b. else if(Expression2)

Statements2;

c. else if(Expression3)

statement3;

d. …

e. else

statement;

f. cout<<statements<<endl;

g. Return 0;

5. }

មានន័យថា៖

- ប្រសិនបើលក្ខខណ្ឌទី១ (Expression1) ពិត នោះ វានឹងដំណើរការ Statement_If
- តែបើ (Expression1) មិនពិត វានឹងពិនិត្យលក្ខខណ្ឌទី២ (Expression2) វានឹងអនុវត្តន៍ Statement_Else-if តាមជំហានបន្តបន្ទាប់នីមួយៗ រហូតទាល់តែជួបលក្ខខណ្ឌចុងក្រោយ
- (Else) ដែលមានន័យថាក្រៅពីលក្ខខណ្ឌខាងលើមិនពិតទាំងអស់ ឬអាចនិយាយបានថាវាមិនពិនិត្យលក្ខខណ្ឌអ្វីទៀតនោះទេ គឺវានឹងអនុវត្តនូវ Statement_Else ចុងក្រោយនេះតែម្តង។

ឧទាហរណ៍៖: ចូរសរសេរជំហាននៃការដោះស្រាយបញ្ហា ដើម្បីបញ្ចូលពិន្ទុ និងវាយតម្លៃនិទ្ទេស ទៅតាមការកំណត់របស់គ្រឹះស្ថានសិក្សា ឬសាលារៀនដូចខាងក្រោម៖

1- និស្សិតដែលបានពិន្ទុទាបជាង ៥០ នោះគឺធ្លាក់
2- និស្សិតដែលបានពិន្ទុយ៉ាងតិច ៥០ ប៉ុន្តែតូចជាង ៦០ នោះគឺជាប់ ទទួលបាននិទ្ទេស E
3- និស្សិតដែលបានពិន្ទុយ៉ាងតិច ៦០ ប៉ុន្តែតូចជាង ៧៥ នោះគឺជាប់ ទទួលបាននិទ្ទេស D
4- និស្សិតដែលបានពិន្ទុយ៉ាងតិច ៧៥ ប៉ុន្តែតូចជាង ៨៥ នោះគឺជាប់ ទទួលបាននិទ្ទេស C
5- និស្សិតដែលបានពិន្ទុយ៉ាងតិច ៨៥ ប៉ុន្តែតូចជាង ៩៥ នោះគឺជាប់ ទទួលបាននិទ្ទេស B

6- និស្សិតដែលបានពិន្ទុយ៉ាងតិច ៩៥ នោះគឺជាប់ ទទួលបាននិទ្ទេស A

1- Start

2- Input Score

3- If (score < 50)

Output (“Fail”)

4- Else-if (score >=50 and score <60)

Output (“Pass Grade E”)

5- Else-if (score >=60 and score <75)

Output (“Pass Grade D”)

6- Else-if (score >=75 and score <85)

Output (“Pass Grade C”)

7- Else-if (score >=85 and score <95)

Output (“Pass Grade B”)

8- Else

Output (“Pass Grade A”)

9- Stop

- យើងអាចធ្វើការប្រៀបធៀបជាមួយនឹងការសរសេរជាមួយនឹងភាសា C++

1- #include<iostream>

2- using namespace std;

3- int score;

4- int main(){

a. if(score<50) cout<<”Fail”<<endl;

b. else if(score>=50 & score<60) cout<<”Pass Grade E”<<endl;

c. else if(score>=60 & score<75) cout<<” Pass Grade D”<<endl;

d. else if(score>=75 & score<85) cout<<” Pass Grade C”<<endl;

e. else if(score>=85 & score<95) cout<<” Pass Grade B”<<endl;

f. else cout<<" Pass Grade A"<<endl;

5- return 0;

6- }

៤.១.៥ ក្បួនដោះស្រាយបញ្ហាដដែលៗ

ក្បួនដោះស្រាយបញ្ហាដដែលៗ ត្រូវបានប្រើដើម្បីធ្វើការងារនោះម្តងហើយម្តងទៀត រហូតទាល់តែទៅជួបលក្ខខណ្ឌទើបឈប់ ឬបញ្ចប់ការងារនោះ។

សម្រាប់ការសរសេរកម្មវិធីវិញ អ្នកសរសេរកម្មវិធីភាគច្រើន មានការលំបាកក្នុងការប្រើប្រាស់ពាក្យថាដដែលៗ ជាងពេលដែលពួកគេប្រើការងារទាក់ទងនឹងការជ្រើសរើស។

ឧទាហរណ៍៖ *ការបោសសម្អាតផ្ទះ* គឺ***បោសរហូតទាល់តែអស់សំរាម***ទើបឈប់។

នៅក្នុងការសិក្សាត្រង់ចំណុចនេះ យើងនឹងធ្វើការបកស្រាយ និងពន្យល់អំពីការងារដដែលៗ ដែលនឹងអនុវត្តន៍ជាមួយក្បួនមួយចំនួនដូចតទៅ។

ការធ្វើការងារដដែលៗ ទាល់តែជួបលក្ខខណ្ឌ (REPEAT-UNTIL) គឺស្រដៀងនឹងរចនាសម្ព័ន្ធនៃការប្រើប្រាស់ដំណោះស្រាយដដែលៗមួយទៀតដែរ គឺការធ្វើការងារដដែលៗ ខណៈពេលដែល (WHILE)។

វាពាក់ព័ន្ធនឹងការប្រតិបត្តិម្តងហើយម្តងទៀត ក៏ពាក់ព័ន្ធនូវលំដាប់លំដាប់លំដោយនៃសេចក្តីណែនាំយោងទៅតាមតម្លៃនៃលក្ខខណ្ឌ។ ទោះយ៉ាងណារចនាសម្ពន្ធ័នៃការធ្វើការងារដដែលៗ ទាល់តែជួបលក្ខខណ្ឌ និងការធ្វើការងារដដែលៗ ខណៈពេលដែល ខុសគ្នាត្រង់ចំណុច៖

- ការធ្វើការងារដដែលៗ ទាល់តែជួបលក្ខខណ្ឌ គឺប្រតិបត្តិតាមការណែនាំដរាបណាជួបនឹងលក្ខខណ្ឌដែលត្រូវបានកំណត់ រីឯការធ្វើការងារដដែលៗ ខណៈពេលដែល នឹងប្រតិបត្តិតាមការណែនាំដរាបណាលក្ខខណ្ឌមិនត្រឹមត្រូវ ឬអាចនិយាយបានម្យ៉ាងទៀតថារចនាសម្ពន្ធនេះនឹងដំណើរការរហូតដល់ការងារក្លាយជាលក្ខខណ្ឌពិតពិត។
- ការធ្វើការងារដដែលៗ ទាល់តែជួបលក្ខខណ្ឌ នឹងពិនិត្យមើលលក្ខខណ្ឌមុនពេលមានការអនុវត្តឡើងវិញ ប៉ុន្តែ ការធ្វើការងារដដែលៗ ខណៈពេលដែល ពិនិត្យស្ថានភាពបន្ទាប់ពីការអនុវត្តឡើងវិញនីមួយៗ ឬអាចនិយាយបានម្យ៉ាងទៀតថា ឯការធ្វើការងារដដែលៗ ខណៈពេលដែល គឺធ្វើការងារមួយសិនមិននឹងពិនិត្យលក្ខខណ្ឌដើម្បីអនុវត្តបន្ត។

៤.១.៥.១ ការធ្វើការងារដដែលៗ ខណៈពេលដែល

ដំណើរការនៃ DO-WHILE ត្រូវបានប្រើដើម្បីបញ្ជាក់រង្វិលជុំដែលមានការសាកល្បង ដោយមានការចាប់ផ្តើមនិងបញ្ចប់ នៃរង្វិលជុំ (Loop) ដែលត្រូវបានបង្ហាញដោយពាក្យគន្លឹះពីរគឺ WHILE និង ENDWHILE ។

ទម្រង់ទូទៅគឺ៖

Expression1;
Do
{
    Statement;
    Expression3;
}while(Expression2)
	WHILE condition
	    sequence
	ENDWHILE

ឧទាហរណ៍៖

```
WHILE ផលិតផលមិនទាន់សម្រេចដល់ការរំពឹងទុក
        ផលិតបន្ថែម
ENDWHILE
```

៤.១.៥.២ ការធ្វើការងារដដែលៗ ទាល់តែជួបលក្ខខណ្ឌ

រង្វិលជុំ (Loop) នេះ គឺប្រហាក់ប្រហែលនឹងរង្វិលជុំ WHILE លើកលែងតែការសាកល្បងត្រូវបានអនុវត្តនៅផ្នែកខាងក្រោមនៃរង្វិលជុំជំនួសនៅខាងលើ។ ពាក្យគន្លឹះពីរ គឺ REPEAT និង UNTIL ត្រូវបានប្រើ។

ទម្រង់ទូទៅគឺ៖

```
REPEAT
        sequence
UNTIL condition
```

ឧទាហរណ៍៖

```
REPEAT
        ការផលិតទំនិញ
```

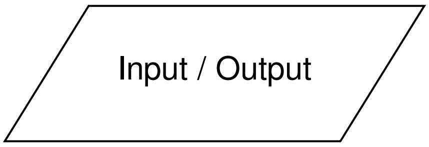
UNTIL ទំនិញគ្រប់គ្រាន់តាមការកំណត់

៤.២ ដំណោះស្រាយបញ្ហាជាគំនូសតាង (Flowchart)

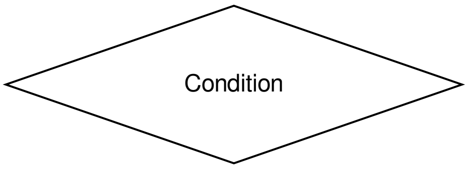
ដំណោះស្រាយបញ្ហាជាគំនូសតាង គឺជាវិធីសាស្រ្ត ឬបច្ចេកទេសទាំងឡាយណាដែលត្រូវបានគេប្រើ ដើម្បីបកស្រាយអំពីចលនការ ទៅតាមបែបបទនៃបញ្ហា ឬសកម្មភាពជាច្រើនដោយបង្ហាញចេញ ដូចគ្នានឹង Algorithms ផងដែរ ដោយគ្រាន់តែបង្ហាញជាគំនូសសញ្ញាផ្សេងៗ ដែលមានរៀងដូចខាងក្រោម៖

សម្រាប់បង្ហាញពីការចាប់ផ្ដើម ឬ បញ្ឈប់ដំណើរការ

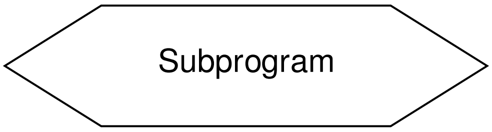
សម្រាប់បង្ហាញដំណើរការនៃការងារ



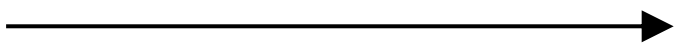
សម្រាប់បង្ហាញពីការបញ្ចូល និងបញ្ចេញទិន្នន័យ



សម្រាប់ត្រួតពិនិត្យមើលលក្ខខណ្ឌ



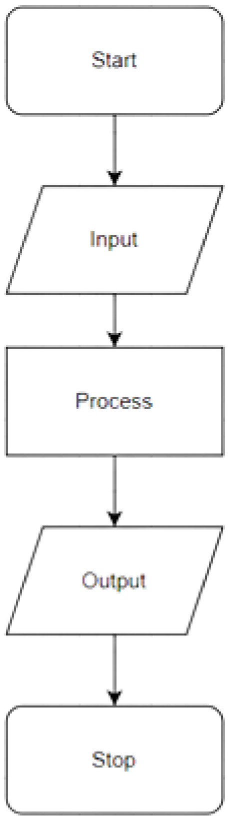
សម្រាប់បង្ហាញពីការរៀបចំ Subprogram


សម្រាប់បង្ហាញពីទិសដៅនៃការ Process

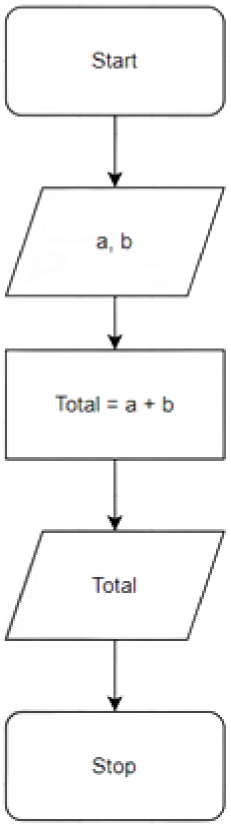
Flowcharts ត្រូវបានបែងចែកចេញជាបីប្រភេទ គឺ ៖

1. គំនូសតាងជាលំហូរការងារ
2. គំនូសតាងលក្ខណៈមានលក្ខខណ្ឌ
3. គំនូសតាងជាលក្ខណៈមានភាពច្រំដែល

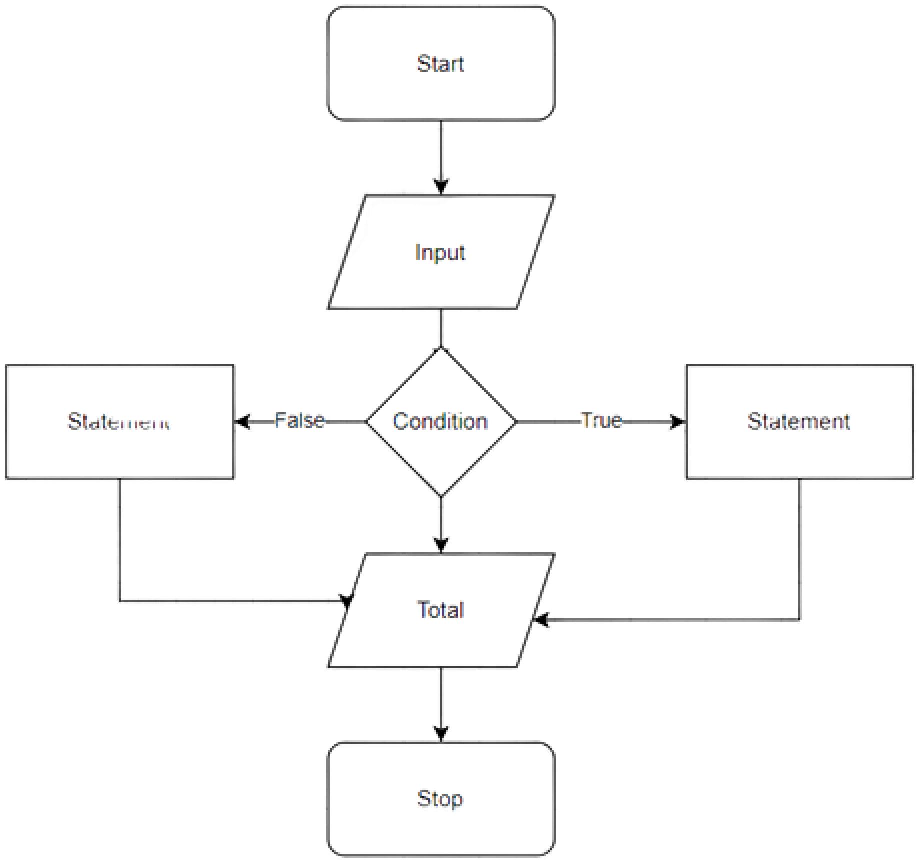
៤.២.១ គំនូសតាងជាលំហូរការងារ



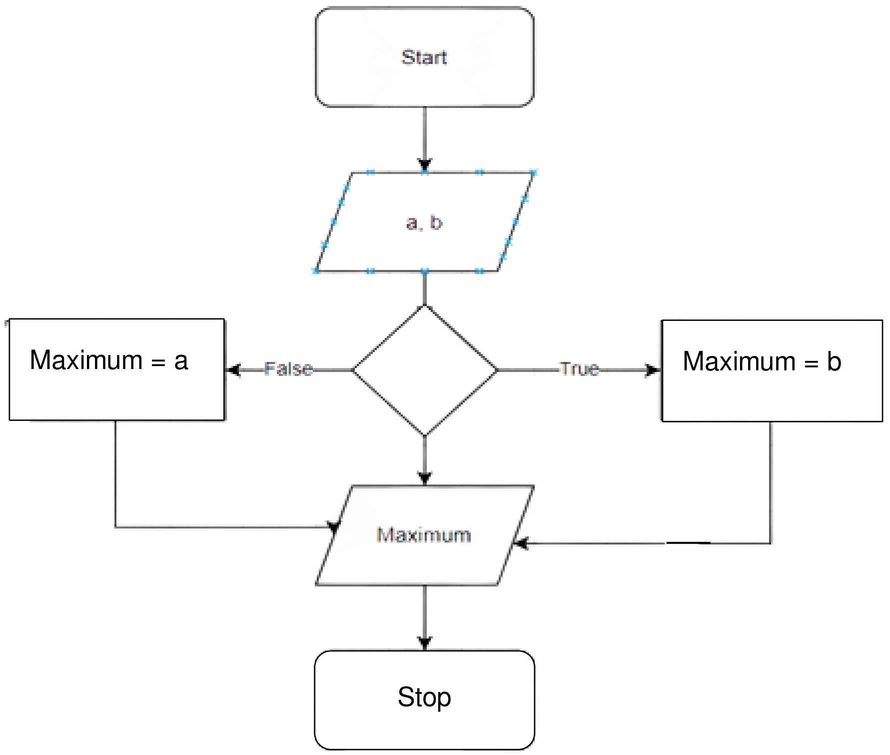
ឧទាហរណ៍៖ ចូរគូសលំហូរការងារសម្រាប់ការបូកពីរចំនួន



៤.២.២ គំនូសតាងជាលក្ខណៈមានលក្ខខណ្ឌ



ឧទាហរណ៍៖ ចូរគូសលំហូរការងារផ្ទៀងផ្ទាត់រវាងពីរតម្លៃ ថាតើតម្លៃណាមួយជាតម្លៃធំជាងគេ

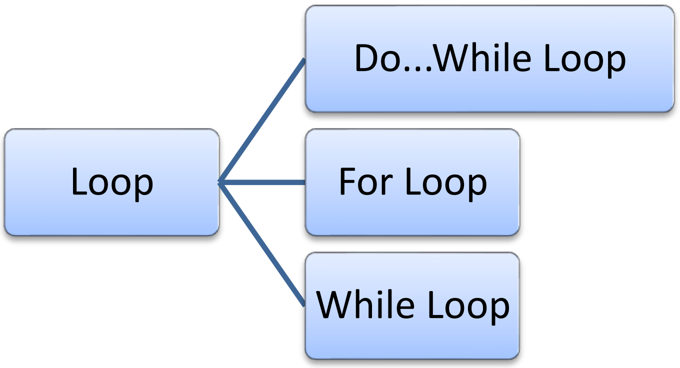


៤.២.៣ គំនូសតាងជាលក្ខណៈមានភាពច្រំដែល ឬធ្វើម្តងហើយម្តងទៀត

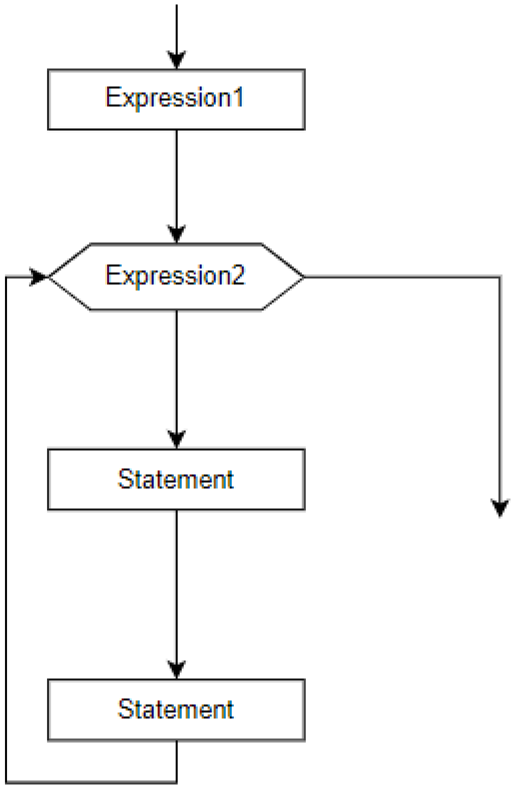
ក្នុងការគូសលំហូរការងារដែលមានភាពច្រំដែល ឬដដែលៗ គឺវាមានលក្ខណៈដូចគ្នាទៅការសរសេរក្បួននៃការដោះស្រាយបញ្ហាដែរ ដោយគ្រាន់តែវាបានលើកយកការដោះស្រាយបញ្ហាដោយធ្វើការពន្យល់នូវលំហូរការងារនោះតាមរយៈរូបនិមិត្តសញ្ញាវិញ។

នៅក្នុងការគូសគំនូសតាងបែបច្រំដែលនេះ យើងនឹងលើកយកនូវវិធីសាស្ត្រចំនួនបី មកធ្វើការពន្យល់ដោយសារការងារនេះ នឹងជួបនៅពេលខាងមុខជាមួយនឹងការសរសេរកូដជាមួយកម្មវិធីជាក់ជាពុំខាន។

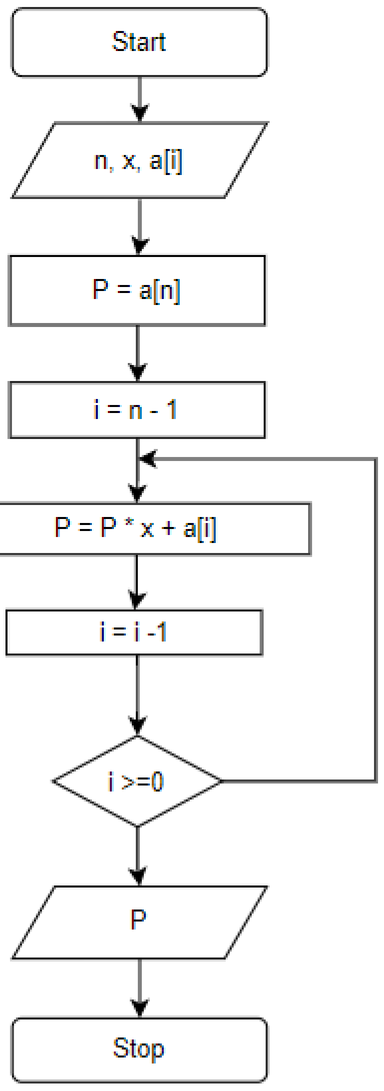
គំនូសតាងការងារដដែលៗ (Loop) ដែលនឹងសិក្សានោះរួមមាន៖



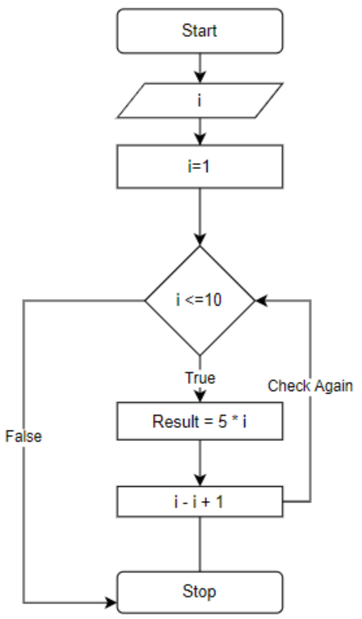
- while loop flowchart



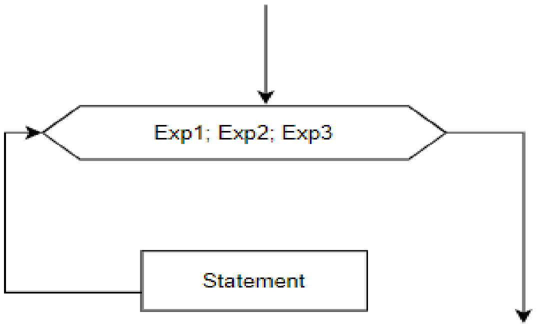
ឧទាហរណ៍៖ ចូរគូសលំហូរប្រតិបត្តិការនៃការបំលែងបព័ន្ធគោលពីរ ទៅជាប្រព័ន្ធគោលដប់



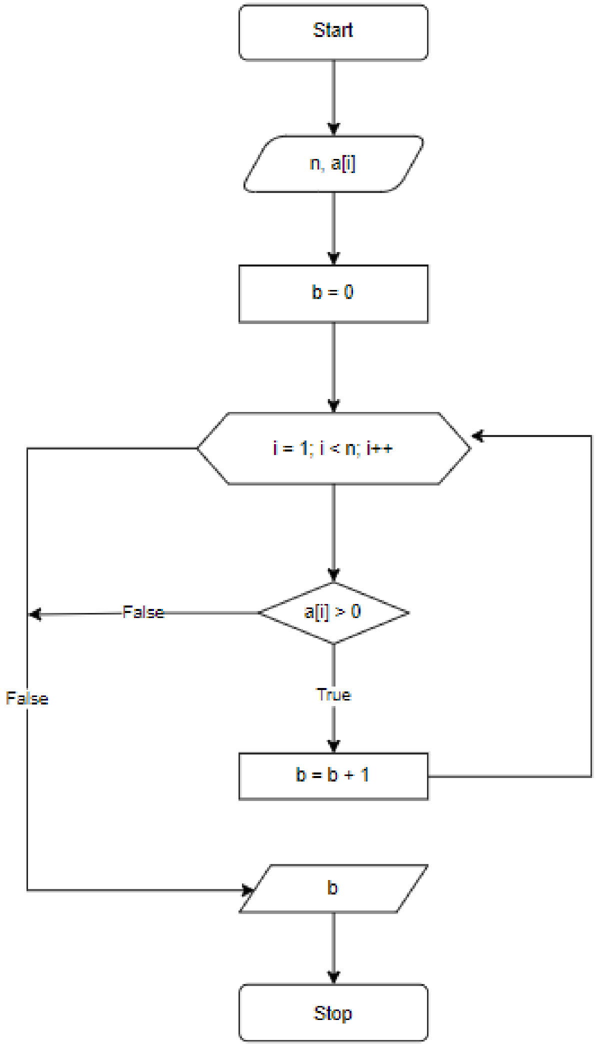
ឧទាហរណ៍៖ ចូរគូសលំហូរការងារនៃការគណនាមេលេខ ៥ ចំនួន ១០ដង



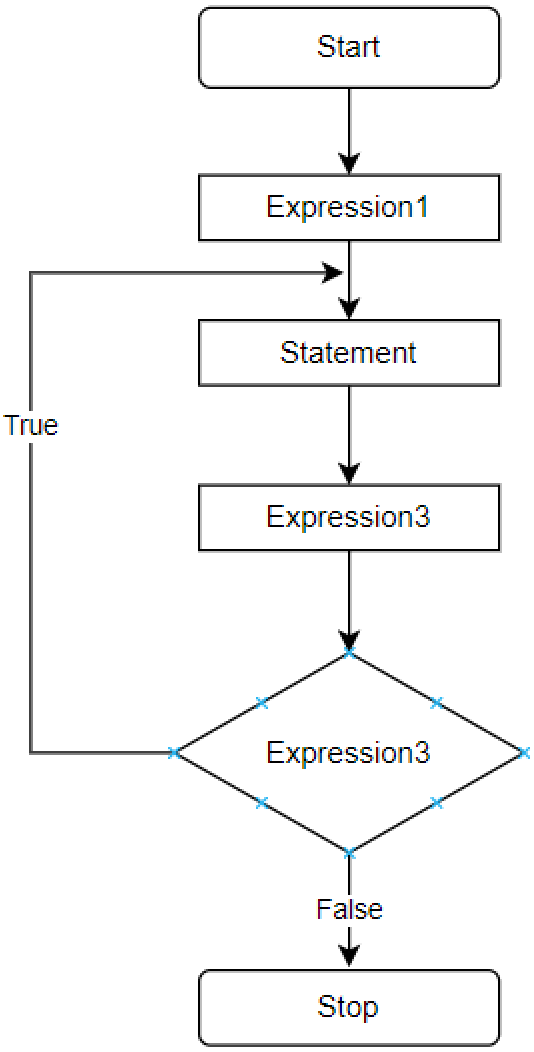
- for… loop flowchart



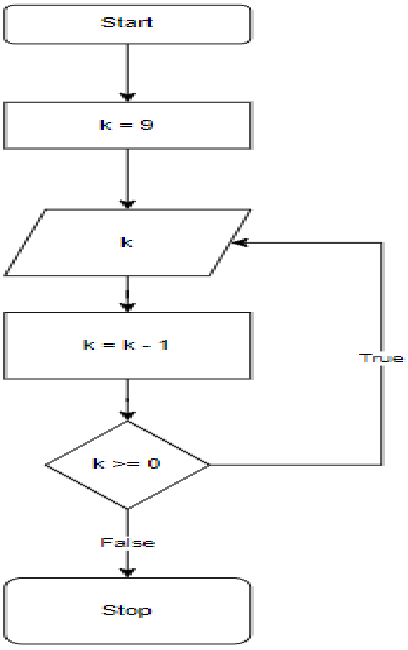
ឧទាហរណ៍៖ ចូរគូសគំនូសតាងសម្រាប់រកតម្លៃវិជ្ជមាន



- do-while… loop



ឧទាហរណ៍៖ ចូរគូសគំនូសតាងលំហូរការងារបង្ហាញតម្លៃលេខចុះពីលេខ ៩ មកដល់លេខ ០



លទ្ធផលនឹងបង្ហាញដូចតទៅ៖

9
8
7
6
5
4
3
2
1
0

- **ចូរសរសេរ Algorithm និងគូស Flowchart**

សម្រាប់ការប្តូរប្រាក់ដែលគាត់មាន គាត់មានប្រាក់មួយចំនួនចង់យកទៅប្តូរ តែមិនដឹងថាគាត់មានប្រាក់ដុល្លារ ឬខ្មែរទេ។ ដោយគិតក្នុងអត្រា 1$=4050៛ និង 4070៛=1$។

- *Algorithm*:

1. Start
2. Input Currency ("Dollar" , "Riel")
3. If (currency = "Dollar")

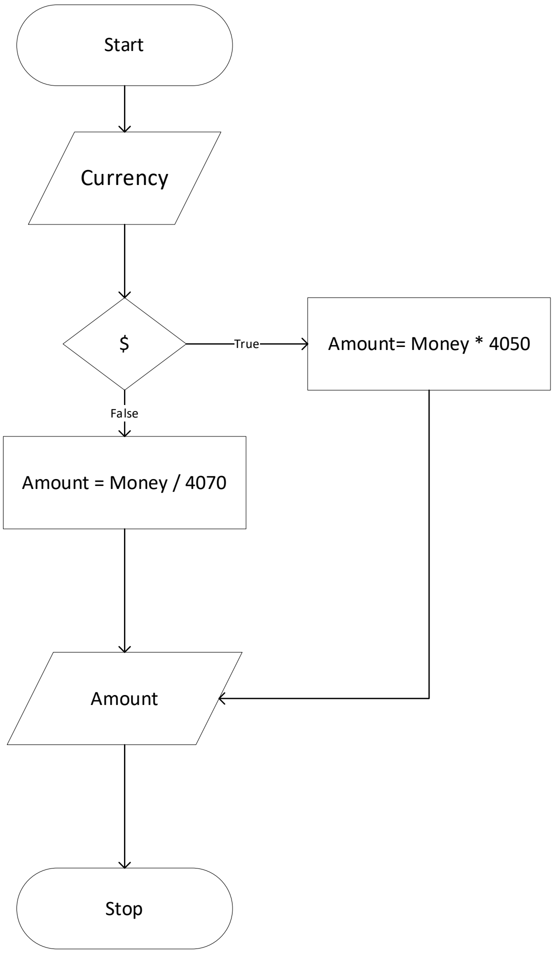
   Amount=Money * 4050

4. Else

   Amount=Money/4070

5. Print Amount
6. Stop

- *Flowchart*



- ចូរសរសេរ Algorithm និងគូស Flowchart ដើម្បីគណនារកប្រាក់ខែរបស់បុគ្គលិកផ្នែកលក់របស់ក្រុមមួយ ដោយក្រុមហ៊ុនបានផ្តល់ប្រាក់លើកទឹកចិត្ត (Commission) ដល់បុគ្គលិកទាំងឡាយណាដែលបានលក់ផលិតផលក្នុងលក្ខខណ្ឌដូចតទៅ៖
  1. Sold តិចជាងឬស្មើ 200$ គឺមិនទទួលបាន Commission នោះទេ
  2. Sold លើសពី 200$ ហើយតិចជាងឬស្មើ 300$ គឺទទួលបាន Commission 20$
  3. Sold លើសពី 300$ ឡើងទៅគឺទទួលបាន Commission 30$ និង 10% នៃប្រាក់ដែលលក់លើសពី 300$

បញ្ជាក់៖ ប្រាក់ខែរបស់បុគ្គលិកម្នាក់ៗ គឺ 200$

- ***Algorithm***
  - Start
  - Input Sold_amount
  - If (Sold_amount <= 200) Then

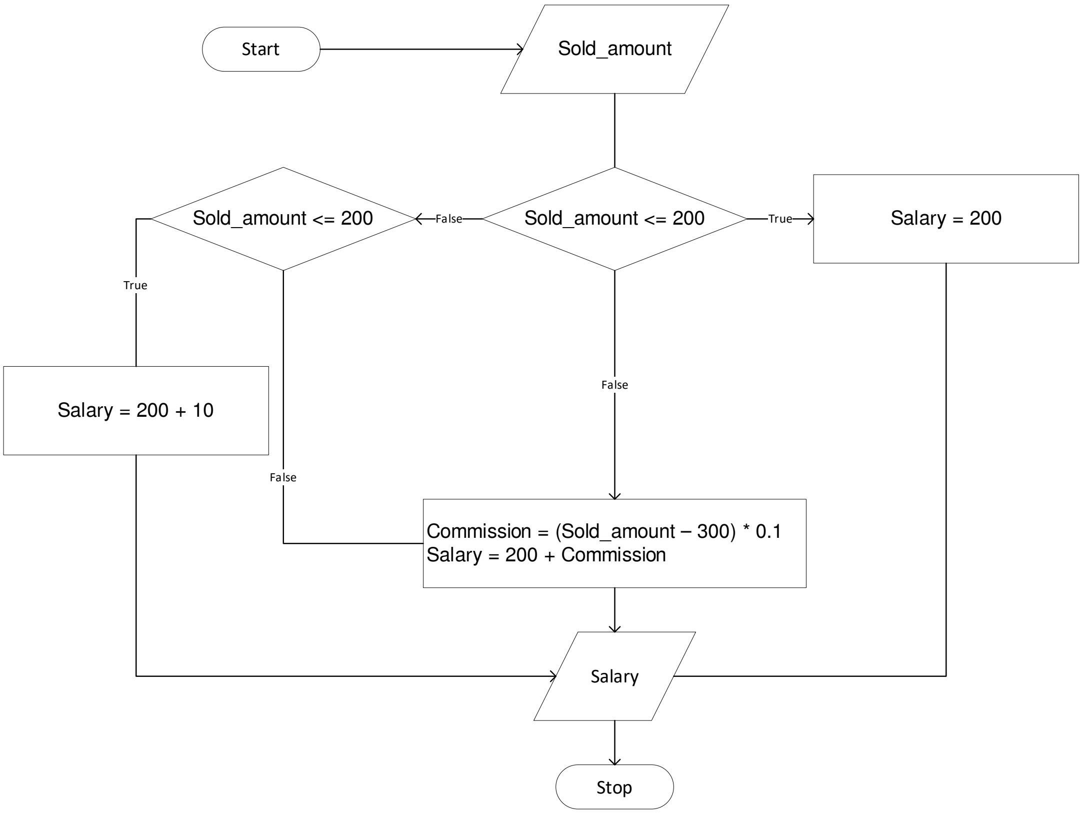
    Salary = 200
  - Else-if (Sold_amount >200 and Sold_amount<=300) Then

    Salary = 200 + 10
  - Else

    Commission = (Sold_amount – 300) * 0.1

    Salary = 200 + Commission
  - Print Salary
  - Stop

- ***Flowchart***



- ចូរសរសេរ Algorithm និងគូស Flowchart ដើម្បីគណនាផលគុណតម្លៃលេខ 2

Algorithm

1- Start

2- Input i, 2

3- Repeat

4- Result = 2 * i

5- While i <=10

6- Print Result

7- Stop

៤.៣ ដំណោះស្រាយបញ្ហាជាភាសាអង់គ្លេស (Pseudo code design language)

ដំណោះស្រាយបញ្ហាជាភាសាអង់គ្លេស គឺជាការបង្កើត Logic ដែលជាភាសាអង់គ្លេស Statement ឬជា clauses ទៅជាជំហាននៃការដោះស្រាយបញ្ហា។

Pseudo Code គឺជាប្រភេទភាសាអង់គ្លេស ដែលមានរចនាសម្ពន្ធសម្រាប់ពិពណ៌នា អំពីក្បួនដោះស្រាយបញ្ហាដោយប្រើប្រាស់ភាសាមនុស្សដើម្បីពន្យល់។ វាអនុញ្ញាតឱ្យអ្នករចនាផ្តោតលើតក្កវិជ្ជា នៃក្បួនដោះស្រាយដោយមិនមានភាពស្មុគស្មាញដោយព័ត៌មានលម្អិតនៃភាសាឡើយ។ ក្នុងពេលជាមួយគ្នានេះដែរ pseudocode ចាំបាច់ត្រូវបំពេញនូវការងារទៅតាមភាពជាក់ស្តែង វានឹងពិពណ៌នាអំពីតក្កវិជ្ជាទាំងមូលនៃក្បួនដោះស្រាយចំណោទបញ្ហា ដូច្នេះការអនុវត្តក្លាយជាភារកិច្ចសំខាន់នៃការបកប្រែនូវជំហាននៃការដោះស្រាយបញ្ហាទៅតាមជួរ ទៅជាភាសាកូដណាមួយ ។

ជាទូទៅវាក្យស័ព្ទដែលត្រូវបានប្រើនៅក្នុងនៃមិនសមរម្យ គួរតែជាវាក្យស័ព្ទនៃដែនបញ្ហាមិនមែនជាដែនអនុវត្តន៍ទេ ដែល pseudocode គឺជាការបកស្រាយបែបនិទានកថា សម្រាប់នរណាម្នាក់ដែលត្រូវដឹងអំពីតម្រូវការ (ដែនបញ្ហា) ហើយកំពុងព្យាយាមរៀនពីរបៀបដែលដំណោះស្រាយត្រូវបានរៀបចំ។

Pseudo code គឺជាប្រភេទនៃ Structure English សម្រាប់ពណ៌នា Algorithms។

ហើយវាក៏អាចលម្អិតតាមជំហានខាងក្រោម៖

- មាននូវភាសាដើមរបស់កម្មវិធី ដូចជា statement ដែលជាក់លាក់ ហើយពណ៌នាអំពីជំហាននៃ Algorithm ឬ program
- statement បង្ហាញពីសកម្មភាព
- លម្អិតទៅលើ logic នៃ algorithm ឬ program
- មិនសង្កត់ធ្ងន់លើភាសានៃកម្មវិធី
- ដោយស្វ័យប្រវត្តិពី Statement

គុណសម្បត្តិនៃ pseudo code គឺ

- វាងាយស្រួលក្នុងការបង្កើតជាមួយនឹងអក្សរ ព្រោះវាប្រើប្រាស់អក្សរអង់គ្លេស Statement
- វាគ្មាននិមិត្តសញ្ញាពិសេសណាមួយសម្រាប់តំណាងនោះទេ វាមានសញ្ញាតិចតួចបំផុតសម្រាប់បញ្ជាក់
- ការងាររបស់វា អាចជាទូទៅបកប្រែនូវមូលដ្ឋានមួយៗ ទៅជាភាសាកម្រិតខ្ពស់
- បញ្ហាដំបូងនៃ Logicសំខាន់បំផុតរបស់ Pseudo coding គឺ កាត់បន្ថយពេលវេលា ក្នុងការសរសេរ កូដ និងបំបាត់កំហុសរបស់ប្រព័ន្ធ

Pseudo Coe ត្រូវបានបែងចែកចេញជា ៩ គឺៈ

1. SEQUENCE Pseudo Code
2. IF-THEN-ELSE Pseudo Code
3. WHILE Pseudo Code
4. CASE Pseudo Code
5. REPEAT-UNTIL Pseudo Code
6. FOR Pseudo Code
7. NESTED CONSTRUCTS Pseudo Code
8. INVOKING SUBPROCEDURES Pseudo Code
9. EXCEPTION HANDLING Pseudo Code

៤.៣.១ SEQUENCE Pseudo Code

SEQUENCE Pseudo Code មានន័យថាជាការងារសរសេរនូវលំហូរការងារតាងឲ្យការងារបែបលក្ខណៈជាលីនេអ៊ែរអនុវត្តជាបន្តបន្ទាប់បន្ទាប់ ពីមួយផ្សេងទៀតដោយគ្មានពិនិត្យលក្ខខណ្ឌណាមួយនោះទេ។

SEQUENCE Pseudo Code គឺនឹងអនុវត្តការងារពីលើចុះក្រោម ឬអាចនិយាយបានថាជាការ អនុវត្តក្នុងការគណនាប្រមាណវិធីសាមញ្ញៗ។

របៀបសរសេរៈ

```
Begin
Set Variable Value
Assign Variable
Process Variable
Print Variable
End
```

ឧទាហរណ៍៖ ចូរសរសេរ Pseudo Code សម្រាប់គណនាផលគុណ ២តម្លៃ

```
Begin
Declare a and b
Assign a=5 and b=10
Set Result to a multiply b
Print Result
End
```

៤.៣.២ IF-THEN-ELSE Pseudo Code

IF-THEN-ELSE Pseudo Code មានន័យថាជាការងារសរសេរនូវលំហូរការងារតាងឲ្យការងារបែបលក្ខណៈជាការផ្ទៀងផ្ទាត់នូវលក្ខខណ្ឌពីរដើម្បីអនុវត្តន៍ការងារ។

របៀបសរសេរ៖

```
Begin
Declare Variable
Assign Variable
If( Variable ) Then Process
Else Process
End
```

ឧទាហរណ៍៖ ចូរសរសេរ Pseudo Code ដើម្បីផ្ទៀងផ្ទាត់ពិន្ទុរបស់និស្សិត ប្រសិនបើតិចជាង ៥0 គឺធ្លាក់ ក្រៅពីនេះគឺជាប់។

```
Begin
Declare Score
Assign Score equal 57
If( Score<50 ) Then Print "Fail"
Else Print "Pass"
End
```

៤.៣.៣ WHILE Pseudo Code

WHILE Pseudo Code មានន័យថាជាការងារសរសេរនូវលំហូរការងារបែបលក្ខណៈជាការផ្ទៀងផ្ទាត់នូវលក្ខខណ្ឌមុននឹងធ្វើការសម្រេចចិត្តដើម្បីអនុវត្តន៍ការងារ។

របៀបសរសេរ៖

```
Begin
Declare Variable
Set Variable to …
While Variable is true Then Statement
        Increase Variable step
        Print Result
End While
End
```

ឧទាហរណ៍៖ ចូរសរសេរ Pseudo Code ដើម្បីអនុវត្តជាមួយនឹងមេលេខ ៥

```
Begin
Declare a
Set a equal 5
Set Count Equal 1
While Count less than or Equal to ten
        Assign Result = a  * Count
        Then Print Result
        Increase Count = Count + 1
End While
End
```

សេចក្ដីពន្យល់៖

បើយើងមើលតាមរយៈការសរសេរ PseudoCode ខាងលើ គឺអាចពន្យល់អ្នកបានថា ការងារដែលវាកំពុងអនុវត្ត គឺគុណមេលេខ ៥ ពី ១ រហូតដល់ ១០។ ក្នុងករណីដែល While ផ្ទៀងផ្ទាត់ថា Count ដែលជាតួត្រូវគុណនោះ នៅតែតូចជាង ឬស្មើ ១០ វានឹងបន្តរហូតដល់ខុសពីលក្ខខណ្ឌទើបវាបញ្ឈប់ការអនុវត្ត បន្ថែមពីចំណុចនេះ សូមអ្នកក្រឡេកមើលត្រង់ចំណុច Increase Count = Count + 1 វាស្ថិតនៅក្នុង While នោះនៅឡើយទេ គឺបានន័យថាវានឹងបង្កើនតម្លៃ Count ពី ១ ទៅជា ២ រួចបន្តទៅផ្ទៀងផ្ទាត់ម្តងទៀត រហូតទាល់តែជួបលក្ខខណ្ឌមិនពិត ឬខុសពីលក្ខខណ្ឌដែលមាននៅខាងលើ វានឹងបញ្ចប់ការអនុវត្ត While នោះហើយក៏បញ្ចប់ការងារនេះផងដែរ។

៤.៣.៤ Case Pseudo Code

Case Pseudo Code មានន័យថាជាការងារសរសេរនូវលំហូរការងារបែបលក្ខណៈជាការផ្ទៀងផ្ទាត់នូវលក្ខខណ្ឌ និងកំណត់យកតាមតម្លៃនីមួយៗដើម្បីអនុវត្តន៍ការងារ។

Case Pseudo Code គឺអាចពន្យល់អ្នកបានថាវានឹងធ្វើការងារទៅតាម Case នីមួយៗដែលមានការកំណត់បានត្រឹមត្រូវ និងមានតាមលក្ខខណ្ឌ។

របៀបសរសេរ៖

```
CASE expression OF
        condition 1 : sequence 1
        condition 2 : sequence 2
        ...
        condition n : sequence n
OTHERS:
        default sequence
ENDCASE
```

ឧទាហរណ៍៖ ចូរសរសេរ Pseudo Code ដើម្បីបង្ហាញនូវរបាយការណ៍ស្តីពីនិទ្ទេសរបស់និស្សិតម្នាក់

```
Begin
Declare Grade
Case based on Grade
        Case >=89 Report "Perfect Score"
        Case > 89    Report "Grade = A"
        Case > 79    Report "Grade = B"
        Case > 69    Report "Grade = C"
        Case > 59    Report "Grade = D"
        Default      Report "Grade = F"
End Case
End
```

សេចក្តីពន្យល់៖

បើយើងមើលតាមរយៈការសរសេរ Pseudo Code ខាងលើ គឺអាចពន្យល់អ្នកបានថា ការងារដែលវាកំពុងអនុវត្ត នៅត្រង់ចំណុច Case based on Grade គឺបានន័យថា ក្នុងករណីដែលដែល Grade មានតម្លៃ

ធំជាង ៨៩ នោះគឺនឹងបង្ហាញថា Perfect Score និងធ្វើការនេះបន្តរហូតដល់ចំណុចចុងក្រោយគឺ Default ដែល អត្ថន័យរបស់ Default គឺមានន័យដូចគ្នានឹង Else ដែរ ដែលមានន័យថាក្រៅពីលក្ខខណ្ឌខាងលើទាំងអស់ វានឹងធ្វើនៅត្រង់នោះ។

៤.៣.៥ REPEAT-UNTIL Pseudo Code

REPEAT-UNTIL Pseudo Code មានន័យថាជាការងារសរសេរនូវលំហូរនៃការងារបែបលក្ខណៈជាការអនុវត្តដដែលៗ គឺអាចនិយាយបានថា ធ្វើម្តងហើយម្តងទៀត រហូតទាល់តែជួបនឹងលក្ខខណ្ឌពិតប្រាកដ ឬលក្ខខណ្ឌដែលបានកំណត់ឲ្យបញ្ចប់ ទើបវាបញ្ឈប់ការងាររបស់វា។

ឧទាហរណ៍៖ អ្នកនឹងទៅដាក់ពាក្យចូលរៀននៅសាកលវិទ្យាល័យមួយ ប៉ុន្តែខ្វះឯកសារ ដូចនេះ អ្នកនឹងត្រឡប់មកយកឯកសារបន្ថែម ឬបើមិនគ្រប់លក្ខខណ្ឌរបស់សាកលវិទ្យាល័យ អ្នកនឹងអនុវត្តការងារនោះរហូតទាល់តែបានសម្រេច។

របៀបសរសេរ៖

```
REPEAT
        sequence
UNTIL condition


Begin
Declare Variable
Set Variable to ..
Repeat
        Print ( Variable )
        Increase/Decrease Variable
Until count = Condition
```

ឧទាហរណ៍៖ ចូរសរសេរ Pseudo Code ដើម្បីអនុវត្តមេលេខ ៨

```
Begin
Declare Number
Set Number=1
Repeat
        Assign Result=8  * Number
```

```
        Increase Number=Number+1
    Until Number=10
    End
```

៤.៣.៦ FOR Pseudo Code

FOR Pseudo Code មានន័យថា ជាការងារសរសេរនូវលំហូរការងារលក្ខណៈជាការអនុវត្ត ធ្វើការងារនោះដដែលៗ ត្រឹមលក្ខខណ្ឌដែលបានកំណត់ដើម្បីឈប់។

របៀបសរសេរ៖

```
    FOR iteration bounds
        sequence
    ENDFOR
```

ឧទាហរណ៍៖

```
    FOR each month of the year ( good )
    FOR month = 1 to 12 ( ok )

    FOR each employee in the list ( good )
    FOR empno = 1 to listsize ( ok )
```

៤.៣.៧ NESTED CONSTRUCTS Pseudo Code

NESTED CONSTRUCTS Pseudo Code មានន័យថាជាការងារសរសេរនូវលំហូរការងារបែបលក្ខណៈមានផ្នែកខាងក្រៅផង និងផ្នែកខាងក្នុងផង ឬអាចនិយាយបានថា វាមានលក្ខណៈលម្អិតត្រង់ថាខាងក្នុងនោះមានលក្ខខណ្ឌពិនិត្យមួយទៀតបន្ថែមពីលក្ខខណ្ឌខាងក្រៅ។

របៀបសរសេរដោយប្រើឧទាហរណ៍៖

```
    SET total to zero
    REPEAT
        READ Temperature
```

```
        IF Temperature > Freezing THEN
            INCREMENT total
        END IF
    UNTIL Temperature < zero
    Print total
```

៤.៣.៨ INVOKING SUBPROCEDURES Pseudo Code

INVOKING SUBPROCEDURES Pseudo Code មានន័យថាជាការងារសរសេរនូវលំហូរការងារបែបលក្ខណៈនៃការហៅអ្វីមួយមកប្រើប្រាស់។

ឧទាហរណ៍៖ ប្រើប្រាស់ពាក្យថា ហៅ (CALL) ដើម្បីទទួលបានលទ្ធផល

```
CALL AvgAge with StudentAges
CALL Swap with CurrentItem and TargetItem
CALL Account.debit with CheckAmount
CALL getBalance RETURNING aBalance
CALL SquareRoot with orbitHeight RETURNING nominalOrbit
```

៤.៣.៩ EXCEPTION HANDLING Pseudo Code

EXCEPTION HANDLING Pseudo Code មានន័យថាជាការងារសរសេរនូវលំហូរការងារបែបលក្ខណៈជំទាស់ ឬបដិសេធការងារណាដែលមិនត្រឹមត្រូវ។

របៀបសរសេរ៖

```
BEGIN
    statements
  EXCEPTION
    WHEN exception type
      statements to handle exception
    WHEN another exception type
      statements to handle exception
  END
```

ឧទាហរណ៍៖

```
BEGIN
    Set age=15
EXCEPTION
    WHEN age < 18
        "You are not allowed to use this method"
    WHEN age >=18
        "You can do what you need"
END
```

លំហាត់

1. ចូរសរសេរ Algorithms ដើម្បីគណនាតម្លៃសរុបនៃការលេងអ៊ិនធើណេត ដោយគេគិតតម្លៃ ២០០០៛ ក្នុង១ម៉ោង
2. ចូរសរសេរ Algorithms ដើម្បីគណនាថ្លៃឈ្នួលរថយន្តក្រុង ដែលធ្វើដំណើរពីភ្នំពេញ ទៅកាន់ខេត្តសៀមរាប ដោយសំបុត្រមួយតម្លៃ ៣០០០០៛ ចុះបើគាត់ទិញសម្រាប់មនុស្សច្រើននាក់ តើត្រូវចំណាយប៉ុន្មាន?
3. ចូរសរសេរ Algorithms ដើម្បីរកតម្លៃសេស ឬគូ នូវលេខដែលគេបានបញ្ចូល
4. ចូរសរសេរ Algorithms ដើម្បីបង្ហាញលទ្ធរបស់និស្សិត ដោយពិន្ទុតិចជាង ៥០ គឺធ្លាក់ បើមិនដូច្នោះទេ គឺជាប់
5. ចូរសរសេរ Algorithms ដើម្បីគណនាថ្លៃប្រើប្រាស់អគ្គិសនី (EDC) ដែលការប្រើប្រាស់នោះមានលក្ខខណ្ឌមួយចំនួនដូចតទៅ
6. ប្រើយ៉ាងច្រើន ១០០KW តម្លៃ ៣៥០៛/ ១KW
7. ចន្លោះពី ១០១KW ដល់ ២០០KW តម្លៃ ៤៥០៛/១KW
8. ចន្លោះពី ២០១KW ដល់ ៣០០KW តម្លៃ ៦០០៛/១KW
9. ក្រៅពីនេះគឺតម្លៃ ៨០០៛/១KW
10. ចូរគូស Flowchart និងសរសេរ Algorithm សម្រាប់រកប្រាក់ពន្ធលើប្រាក់ខែ ការយកពន្ធលើប្រាក់ខែរបស់បុគ្គលិកម្នាក់ៗ ដែលបានកំណត់ដូចខាងក្រោម៖
    a. បើប្រាក់ខែនៅចន្លោះ $1000 ទៅ $1800 នោះគេដកយកពន្ធ $100 ហើយថែមជូន 1% លើប្រាក់ដែលលើសពី $1000
    b. បើប្រាក់ខែតិចជាង $2500 នោះគែដកយកពន្ធ $200 ហើយថែម 1.5% លើប្រាក់ដែលលើសពី $2000
11. ចូរសរសេរ Algorithm ដើម្បីសរុបតម្លៃនៃការយាមម៉ូតូ ដោយគេដឹងថា ផ្ញើម៉ូតូមុនម៉ោង ១៥ គេយកតម្លៃ ៥០០៛ ក្នុង ១ម ហើយបើផ្ញើក្រោយពីនេះ គឺយកតម្លៃ ១០០០៛ ក្នុង ១ម។
12. ចូរគូស Flowchart និងសរសេរ Algorithms ដើម្បីគណនាតម្លៃនៃការប្រើប្រាស់អគ្គិសនីរបស់អតិថិជន ដែលមានការគណនាដូចខាងក្រោម៖
    a. បើប្រើតិចជាង 100kw នោះគិតក្នុងតម្លៃ ៣៥០៛ ក្នុង ១kw
    b. ប្រើពី 101kw ទៅដល់ 200kw នោះគិតក្នុងតម្លៃ ៤០០៛ ក្នុង ១kw

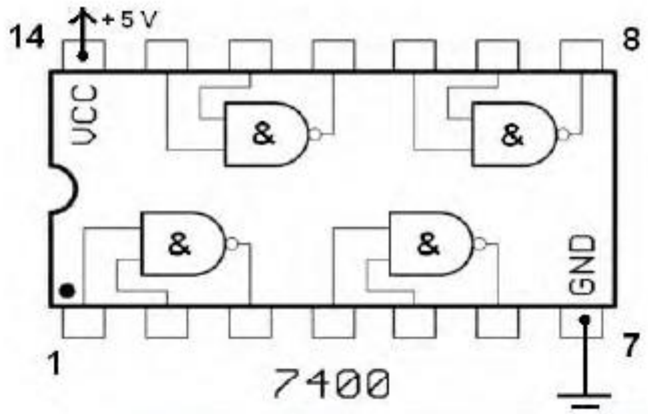
    C. និងលើសពីនេះគឺ ៥៥០៛ ក្នុង ១kw

មេរៀនទី៥ ពិជគណិតពិត-មិនពិត

គោលបំណង

- អាចប្រើប្រាស់បានជាមួយនឹង AND, OR, NOT
- អាចប្រើប្រាស់រួមបញ្ចូលគ្នាជាលក្ខណៈ Circuit
- យល់ដឹងអំពីភាពស្មើគ្នានៃ circuits
- យល់ដឹងអំពីច្បាប់នៃ Boolean Algebra

- ស្ថានភាពនៃ Logic តាងដោយ 0 និង 1
- និមិត្តសញ្ញាផ្សេងៗត្រូវបានតាងដោយ 0 និង 1 ទៅតាមលក្ខខណ្ឌ



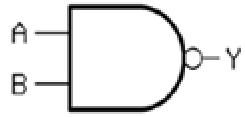
និមិត្តសញ្ញានៃ Logic

ដូចដែលបានរៀបរាប់នៅក្នុងការពិសោធន៍ពីមុនច្រក NAND និង NOR អាចត្រូវបានសាងសង់ដោយមានសមាសធាតុតិចជាងច្រក AND និង OR ។

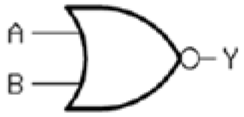
សម្រាប់ហេតុផលនេះអាំងវឺតទ័រ NAND និង NOR បង្កើតជាបន្ទះឈីបចំនួនបួនក្នុងចំណោមប្រាំពីរដែលប្រើនៅក្នុងការពិសោធន៍ទាំងអស់។

និមិត្តសញ្ញាដែលត្រូវបានប្រើ ដើម្បីតំណាងឱ្យ NAND, NOR, AND, OR និងអាំងវឺតទ័រ ឬមិនត្រូវបានផ្តល់ជូនរួមជាមួយតារាងការពិតសម្រាប់ NAND និង NOR ។ តារាងភាពពិតមាន " 0" តំណាងឱ្យមិនពិតនិង " ១ " តំណាងអោយការពិត។

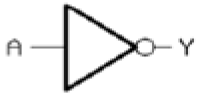
សៀគ្វីដែលអាចត្រូវបានប្រើដើម្បីបង្កើតច្រកទ្វារ NAND ដោយប្រើត្រង់ស៊ីស្ទ័រពីរត្រូវបានបង្ហាញនៅក្នុងវត្ថុតាង ២.៧ ។ ការកំណត់សៀគ្វីសម្រាប់ច្រកទ្វារ NAND ដែលផ្តល់ដោយបន្ទះឈីបស៊េរី ៧៤០០ ខណៈដែលសមមូលឡូជីខលខុសគ្នាពីការរចនានេះ។



NAND



NOR



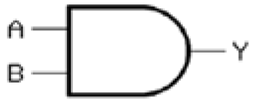
Inverter

A	B	Y
0	0	1
0	1	1
1	0	1
1	1	0

NAND Table

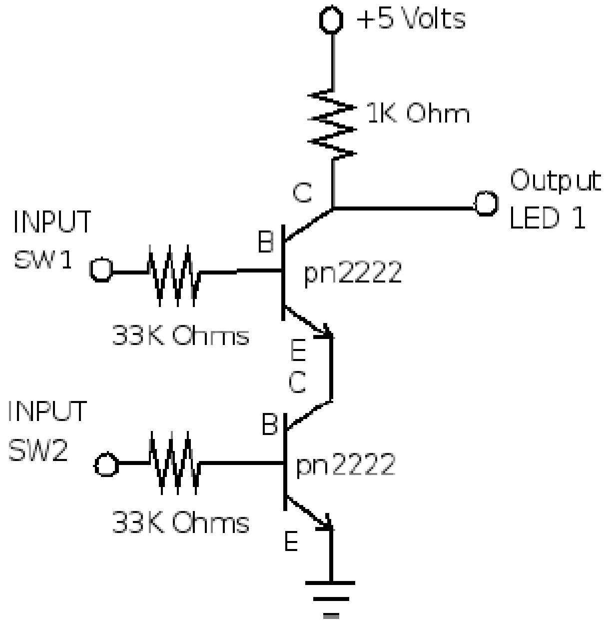
A	B	Y
0	0	1
0	1	0
1	0	0
1	1	0

NOR Table



AND

OR



NAND សៀគ្វី

៥.១ តារាងភាពពិតនៃអនុគមន៍

បើអនុគមន៍មាន n អថេរ នោះតារាងភាពពិតរបស់អនុគមន៍នោះ គឺជាសញ្ញាផ្សេងៗដែលអាចមានតម្លៃ (n+1) ជួរ និង 2^n បន្ទាត់។

ឧទាហរណ៍ៈ ចំពោះ n=2 អនុគមន៍មានពីរអថេរ: f(A, B) = A.B

Input		Output
A	B	f(A, B) = A.B
1	1	1
1	0	0
0	1	0
0	0	0

៥.២ អនុគមន៍ Logic ដំបូង

អនុគមន៍ Logic ដំបូងមានតែ ៣ ប៉ុណ្ណោះគឺ

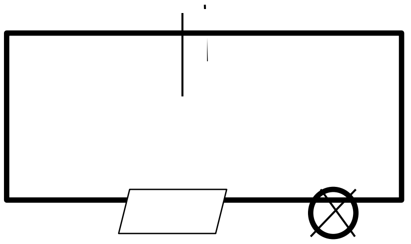
1. អនុគមន៍ Logic និង (AND) មានន័យថា ផលគុណ Logic
2. អនុគមន៍ Logic ឬ (OR) មានន័យថា ផលបូក
3. អនុគមន៍ Logic មិន (NOT)

៥.២.១ អនុគមន៍ Logic និង (AND)

អនុគមន៍ Logic និង ប្រើសម្រាប់ភ្ជាប់អថេរ ២ ឬច្រើន ហើយវាមានតម្លៃស្មើ ១ ក្នុងករណីដែលយើង input 1 ឡើង

Input		Output
A	B	f(A, B) = A AND B
1	1	1
1	0	0
0	1	0
0	0	0

បកស្រាយតាម Circuit អគ្គិសនីបានថា



តាមរូបគេបាន

- បើអាំងទែរុបទ័រ A បិទ => A=1
- បើអាំងទែរុបទ័រ B បិទ => B=1
- អំពូលភ្លឺកាលណា A និង B បិទទាំងអស់គ្នា

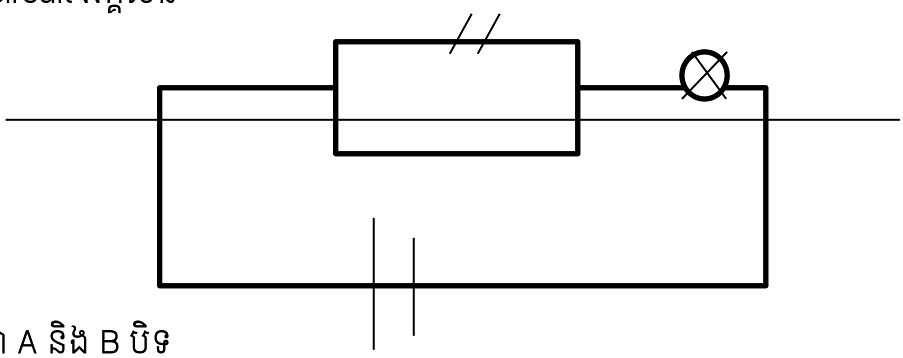
៥.២.២ អនុគមន៍ Logic ឬ or

ប្រើសម្រាប់ភ្ជាប់អថេរពីរ ឬច្រើន ហើយវាមានតម្លៃស្មើ ១ យ៉ាងតិច។

តារាងភាពពិត

Input		Output
A	B	f(A, B) = A + B
1	1	1
1	0	1
0	1	1
0	0	0

បកស្រាយតាម Circuit អគ្គិសនី



អំពូលភ្លឺកាលណា A និង B បិទ

៥.២.៣ អនុគមន៍ logic មិន

វាគឺជាអនុគមន៍នៃអថេរមួយ ដែលមាននិមិត្តរូប (–) អានថា បារ ដាក់លើអថេរ។

តារាងភាពពិត

Input	Output
A	f(A)= $\bar{A}$
1 0	0 1

៥.៣ លក្ខណៈនៃអនុគមន៍ Boolean

1. លក្ខណៈទី ១

   A+0 =A (0 ជាធាតុណឺតចំពោះ +)

   A+1=A (0 ជាធាតុណឺតចំពោះ x)

2. លក្ខណៈត្រឡប់

   A+B=B+A

   A.B=B.A

3. លក្ខណៈផ្តុំ

   A + (B + C) = (A+B) +C

   A. (B.C) = (A.B).C

4. លក្ខណៈបំបែក

   A + (A+B) = AB + AC

   A + (B.C) = (A+B)(A+C)

5. លក្ខណៈបំពេញ

$A + \bar{A} = 1$
$A . \bar{A} = 0$

លក្ខខណ្ឌដូចដែលបានរៀបរាប់ខាងលើ វាជាមូលដ្ឋានសម្រាប់ប្រើ ដោះស្រាយ ឬស្រាយបញ្ជាក់ទ្រឹស្តីបទរបស់ Boolean Algebra។

៥.៤ គោលការណ៍ទ្វេភាពរបស់ Boolean Algebra

រាល់អនុគមន៍ Logic នៅតែជាភាពពិតដដែល កាលណាគេជំនួស (+) ដោយ (*) និង 0 ដោយ 1 ហើយបើគេជំនួស (*) ដោយ (+) និង 1 ដោយ 0។

អនុគមន៍ Boolean មានទម្រង់មួយចំនួនដូចខាងក្រោម៖

- ផលបូកនៃផលគុណ

  ឧទាហរណ៍៖: $f(x_1, x_2, x_3) = (x_1 x_2) + (x_2 x_3)$
- ការភ្ជាប់

  ឧទាហរណ៍៖: $f(x_1, x_2, x_3) = (x_1 + x_2 + x_3)(x_2 + x_3)$
- ទម្រង់ធម្មតា

នៅក្នុងទម្រង់នេះកាលណាតួនីមួយៗ មានអថេរទាំងអស់ នៅក្នុងទម្រង់នេះមាន ២ ទៀតគឺ

<u>ទម្រង់នៃផលបូកនៃផលគុណធម្មតា</u>

ឧទាហរណ៍៖: $f(x_1, x_2, x_3)$ ដែលបង្ហាញបានដោយតារាងដូចខាងក្រោមនេះ៖

Input	Output
x_1, x_2, x_3	$f(x_1, x_2, x_3)$
1 1 1	1
1 1 0	0
1 0 1	1
1 0 0	1
0 1 1	0
0 0 1	1
0 0 0	0

ឧទាហរណ៍៖ $f(x_1, x_2, x_3)$ ដែលបង្ហាញដោយតារាងខាងក្រោមនេះ៖

Input			Output
x_1	x_2	x_3	$f(x_1, x_2, x_3)$
1	1	1	1
1	1	0	0
1	0	1	1
1	0	0	1
0	1	1	0
0	0	1	1
0	0	0	0

៥.៥ ការបង្រួមអនុគមន៍ Boolean

ការបង្រួមអនុគមន៍វាគឺជាអនុគមន៍ទៅលើលក្ខណៈគ្រឹះ និងទ្រឹស្តីបទ ដែលធ្វើតាមវិធីផ្សេងៗ ដែលយើងអាចបានសិក្សារួចមកហើយ។

ឧទាហរណ៍:

៥.៦ ជាតុបំពេញនៃអនុគមន៍ Boolean

ជាតុបំពេញនៃអនុគមន៍ F និង $\bar{F}$ ដែលមានការប្តូរពី 1 ទៅ 0 និងពី 0 ទៅ 1។

តារាងភាពពិតនៃអនុគមន៍ Boolean

Input			Output	
x	y	z	F	$\bar{F}$
1	1	1	0	1
1	1	0	1	0
1	0	1	0	1
1	0	0	1	0
0	1	1	1	0
0	1	0	1	0

0	0	1	0	1
0	0	0	0	1

នៅក្នុង Boolean Algebra ជាតុបំពេញរបស់អនុគមន៍ Boolean អាចទាញតាមទ្រឹស្តីបទ Morgan ដែលគេអាចសរសេរដូចខាងក្រោម៖

$\overline{A_1A_2 \dots A_n} = \bar{A}_1 + \ \bar{A}_2 + \cdots + \ \bar{A}_n$

$\bar{A}_1 + \ \bar{A}_2 + \cdots + \ \bar{A}_n = \overline{A_1A_2 \dots A_n}$

៥.៧ ទម្រង់ Minterm និង Maxterm

- Minterm គឺជា term មួយរបស់ផលបូកនៃផលគុណធម្មតា
- Maxterm គឺជា term មួយរបស់ភាពធម្មតានៃការតភ្ជាប់

ឧទាហរណ៍៖

Input			Output	
x	y	z	Min	Max
1	1	1	xyz	$\bar{x}$ + $\bar{y}$ + z
1	1	0	xyz	$\bar{x}$ + $\bar{y}$ + z
1	0	1		$\bar{x}$ + y + z
1	0	0		x + $\bar{y}$ + z
0	1	1		x + $\bar{y}$ + z
0	1	0		x + $\bar{y}$ + z
0	0	1		x + y + z
0	0	0		x + y + z

៥.៨ ផលបូកនៃផលគុណ

ទម្រង់ផលបូកនៃផលគុណជាផលបូកនៃ term មួយ ឬច្រើន ឬផលបូក នៃ term ច្រើន នៃតក្កវិទ្យា។

ដើម្បីងាយស្រួលក្នុងការបំបែកអនុគមន៍មួយទៅក្នុងទម្រង់មួយ ប្រសិនបើអនុគមន៍នោះមិនស្ថិតក្នុងទម្រង់ផលបូកយើងត្រូវពិនិត្យតួមួយៗ ថាតើវាមានអថេរទាំងអស់ទេ បើខ្វះមួយ ឬច្រើនយើង ត្រូវគុណតួនោះជាមួយនឹងកន្សោម (x + y′)។

ឧទាហរណ៍៖ ពន្លាត A + BC ទៅក្នុងទម្រង់ផលបូកនៃផលគុណ

A(B + B′) = AB + AB′

៥.៩ ផលគុណនៃផលផលបូក

- សង់តារាងភាពពិតដើម្បីធ្វើជាអនុគមន៍ Boolean
- ទម្រង់ Maxterm ដែលបានមកពីការផ្សំរវាងតួដែលមានផលបូកស្មើសូន្យ
- យកកន្សោមផលគុណ Maxterm ដែលបានពីជំហានទីពីរ។

តារាភាពពិត

Input			Output
A	B	C	F(A, B, C)
1	1	1	0
1	1	0	1
1	0	1	0
1	0	0	0
0	1	1	1
0	1	0	1
0	0	1	0
0	0	0	0

តាមតារាងភាពពិតគេទទួលបានអនុគមន៍ f(A, B, C) បង្ហាញតម្លៃស្មើ 0 រួមមាន៖

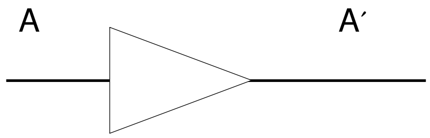
111 101 100 001 000

៥.១០ Logic Circuit (gate)

សញ្ញាប្រមាណវិធីទាំងអស់ដែលប្រើក្នុង Compute ត្រូវបានអនុត្តដោយការផ្សំគ្នា ដែលភ្ជាប់គ្នាដោយកំណត់ជា Block ដែលបង្កើតជា Circuit ហើយហៅថា logic circuit ហើយម្យ៉ាងទៀត Logic Circuit គឺជា Electronic circuit ដែលប្រតិបត្តិមួយ ឬច្រើនក្នុងការបញ្ចូលដើម្បីបានលទ្ធផលខាងក្រៅ។ Logic circuit ទាំងនោះបង្កើតឡើងដើម្បីប្រើក្នុង computer។

៥.១០.១ NOT gate

NOT gate គឺជាការផ្សំដែលបានមកពីប្រមាណវិធីជាគុបំពេញ ហើយម្យ៉ាងទៀត Not gate អាចហៅថា inverter ព្រោះវាបានមកពីការផ្ទុយនៃតម្លៃដែលបញ្ចូល។ NOT gate ត្រូវបានតាងដោយសញ្ញាខាងក្រោម៖



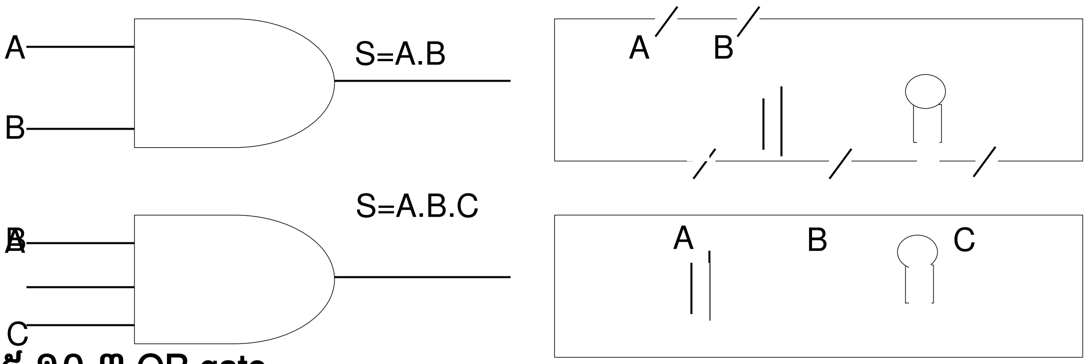
Input	Output
A	A´
1	0
0	1

៥.១០.២ AND gate

AND gate ជាការផ្សំមកពីសញ្ញាប្រមាណវិធីផលគុណតក្កវិទ្យាហើយគឺជា Electronic Circuit ដែលមានតម្លៃស្មើ 1 កាលណាយើងបញ្ចូលតម្លៃស្មើ 1 ផងដែរ។

គេមានតារាងភាពពិត និងនិមិត្តសញ្ញាដូចខាងក្រោមនេះ៖

Input		Output
A	B	A´
1	1	1
0	0	0
1	0	0
0	1	0

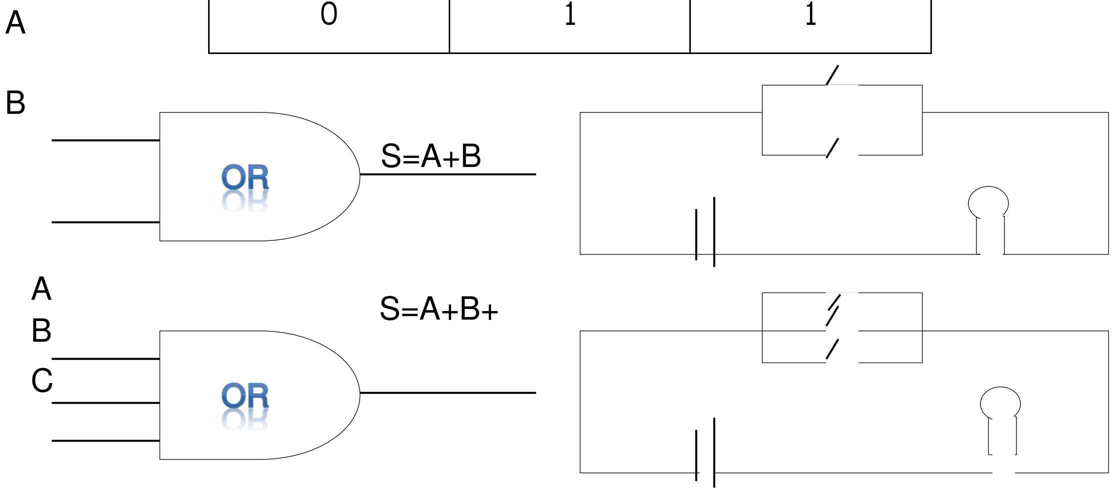


៥.១០.៣ OR gate

វាជាការផ្សំដែលបានមកពីប្រមាណវិធីបូកតក្កវិទ្យា ហើយវាគឺជា Electronic circuit ដែលមានតម្លៃស្មើ 1 កាលណាយើងបញ្ចូលតម្លៃស្មើ 1 យ៉ាងតិចក្នុងនោះ។

យើងមានតារាងភាពពិតដូចខាងក្រោម៖

Input		Output
X	Y	F(X, Y)
1	1	1
0	0	0
1	0	1
0	1	1

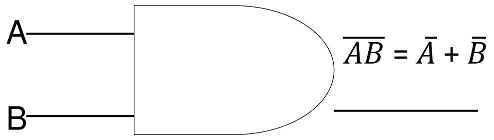


៥.១០.៤ NAND gate

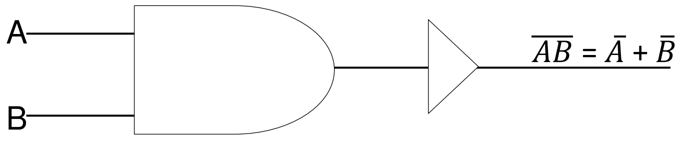
វាជាធាតុបំពេញរបស់ AND gate ដែលវាមានតម្លៃស្មើ 1 កាលណាបញ្ចូលតម្លៃ 0។

តារាងភាពពិត និងនិមិត្តសញ្ញា

Input		Output
X	Y	F(X, Y)
1	1	0
0	0	1
1	0	1
0	1	1



សញ្ញា NAND gate អាចវិភាគដោយសមមូលទៅនឹង Block circuit ដែលមាន AND gate ហើយនិង NOT gate។



៥.១០.៥ NOR gate

NOR gate គឺជាធាតុមួយនៃ OR gate ហើយវាជា Electronic ដែលមានតម្លៃស្មើ 1 ក្នុងករណីមួយគត់ដែលយើងបញ្ចូលតម្លៃ 0។

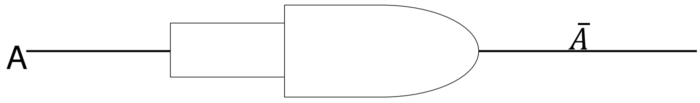
តារាងភាពពិត និងនិមិត្តសញ្ញា

Input			Output
X	Y	Z	
1	1	1	0
1	1	0	0
1	0	1	0
1	0	0	0
0	1	1	0
0	1	0	0
0	0	1	0
0	0	0	1

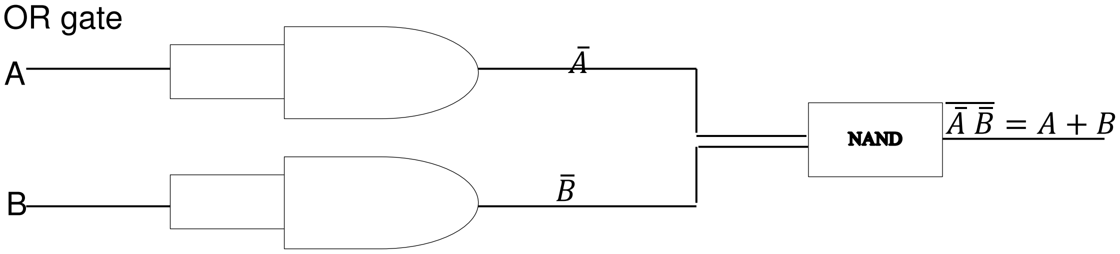
៥.១១ បណ្តុំនៃ NAND gate

វាជាការប្រតិបត្តិខ្លួនឯងដើម្បីទាញឱ្យបាន AND, OR, NOT gate។

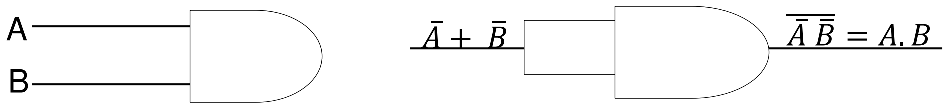
1. NOT gate



2. OR gate



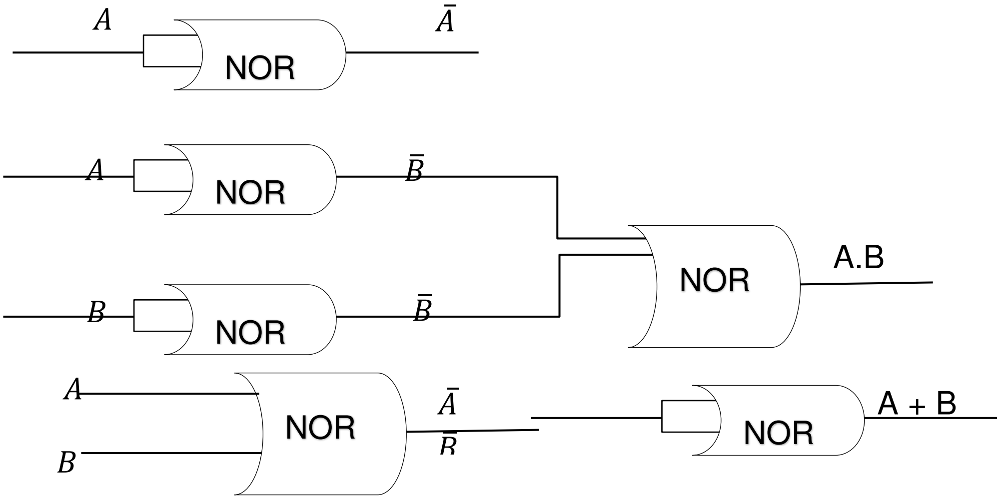
3. AND gate



ចំណាំ៖ NAND ទៅជា NAND gate គឺជាទូទៅវាស្មើទៅនឹង AND និង OR gate។

៥.១២ បណ្តុំនៃ NOR gate

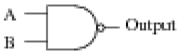
វានឹងប្រតិបត្តិខ្លួនឯងដើម្បីទាញឱ្យបានឃើញប្រមាណវិធី AND, OR, NOT gate។



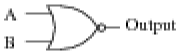
ចំណាំ៖ NOR ទៅជា NOR gate គឺជាទូទៅវាស្មើទៅនឹង OR ទៅជា AND gate។

លំហាត់

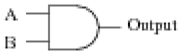
1. ចូរកំណត់នូវរាល់ដំណើរការក្នុងតារាងភាពពិតខាងក្រោម



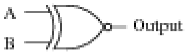
A	B	Output
0	0	
0	1	
1	0	
1	1	



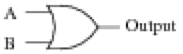
A	B	Output
0	0	
0	1	
1	0	
1	1	



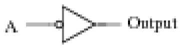
A	B	Output
0	0	
0	1	
1	0	
1	1	



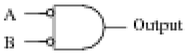
A	B	Output
0	0	
0	1	
1	0	
1	1	



A	B	Output
0	0	
0	1	
1	0	
1	1	



A	Output
0	
1	

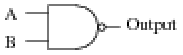


A	B	Output
0	0	
0	1	
1	0	
1	1	

A	B	Output
0	0	
0	1	
1	0	
1	1	

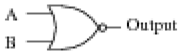
A	B	Output
0	0	
0	1	
1	0	
1	1	

NAND



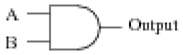
A	B	Output
0	0	1
0	1	1
1	0	1
1	1	0

NOR



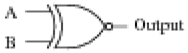
A	B	Output
0	0	1
0	1	0
1	0	0
1	1	0

AND



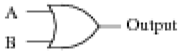
A	B	Output
0	0	0
0	1	0
1	0	0
1	1	1

XNOR



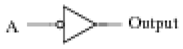
A	B	Output
0	0	1
0	1	0
1	0	0
1	1	1

OR



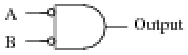
A	B	Output
0	0	0
0	1	1
1	0	1
1	1	1

NOT



A	Output
0	1
1	0

Neg-AND

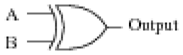


A	B	Output
0	0	1
0	1	0
1	0	0
1	1	0

Neg-OR

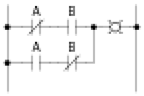
A	B	Output
0	0	1
0	1	1
1	0	1
1	1	0

XOR

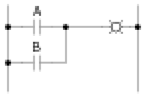


A	B	Output
0	0	0
0	1	1
1	0	1
1	1	0

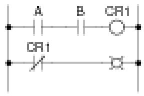
2. ចូរកំណត់នូវតម្លៃនៃប្រកតក្កដូចជា (AND, OR, NOR, etc.) និងគណនាតាមតារាងមក បញ្ជាក់ផង



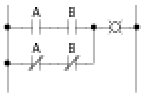
A	B	Output
0	0	
0	1	
1	0	
1	1	



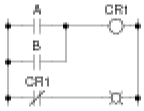
A	B	Output
0	0	
0	1	
1	0	
1	1	



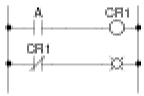
A	B	Output
0	0	
0	1	
1	0	
1	1	



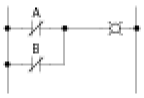
A	B	Output
0	0	
0	1	
1	0	
1	1	



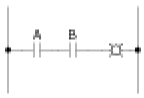
A	B	Output
0	0	
0	1	
1	0	
1	1	



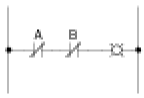
A	Output
0	
1	



A	B	Output
0	0	
0	1	
1	0	
1	1	



A	B	Output
0	0	
0	1	
1	0	
1	1	



A	B	Output
0	0	
0	1	
1	0	
1	1	

ឯកសារយោង

[1]. Mathematics for Computer Science revised Monday 18th May, 2015, 01:43.

[2]. https://users.csc.calpoly.edu/~jdalbey/SWE/pdl_std.html

[3]. https://robertsj.github.io/me400_notes/lectures/Algorithms_Flowcharts_and_Pseudocode.html#Sequence-by-Example:-Computing-Grades

[4]. https://www.programiz.com/dsa/algorithm

[5]. https://ocw.mit.edu/courses/electrical-engineering-and-computer-science/6-042j-mathematics-for-computer-science-fall-2005/lecture-notes/

គាំទ្រលើថវិការៀបរៀង និពន្ធ និងកែលម្អដោយ
ក្រសួងអប់រំ យុវជន និងកីឡា

មូលនិធិស្រាវជ្រាវ គំនិតច្នៃប្រឌិត និងនវានុវត្តន៍