

មេរៀនទី១

សេចក្តីផ្តើមពី Programming

រៀបចំដោយ៖ សេង សឹង

Tel: 093 77 12 44

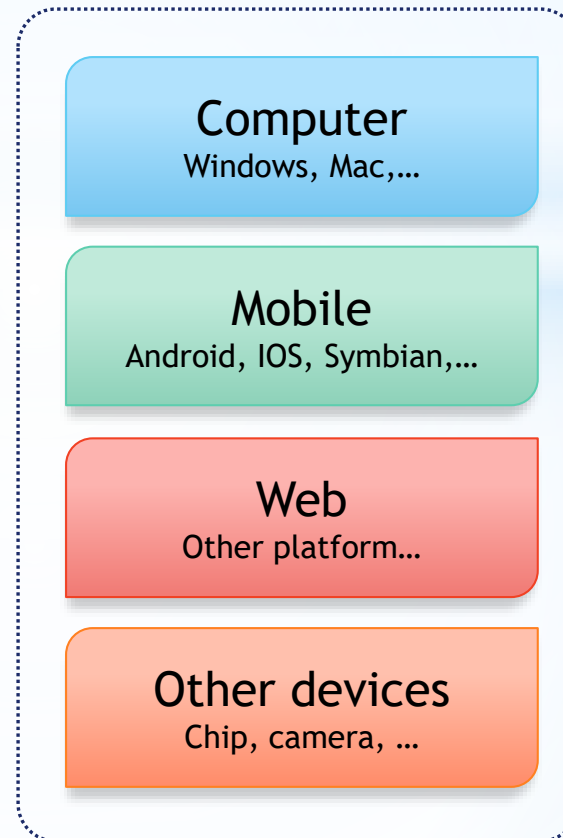
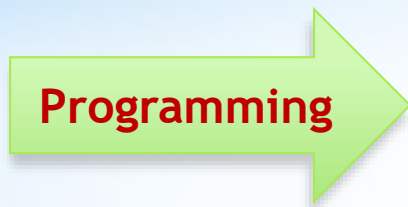
Email: sengsourng@gmail.com

Programming ជាអ្វី?

- វាពិតជាមានសារៈសំខាន់បំផុតដែលអ្នកត្រូវតែស្វែងយល់អំពី ប្រភេទនៃមុខវិជ្ជាក៏ដូចជា អត្ថប្រយោជន៍នៃមុខវិជ្ជាដែលអ្នកកំពុងរៀន។
- Computer Programming គឺសំដៅទៅលើ ដំណើរការរៀបចំនូវដំណោះស្រាយបញ្ហា ការងារទាំងឡាយដែលកើតមានប្រចាំថ្ងៃ មកអនុវត្តនៅលើ ប្រព័ន្ធប្រព័ន្ធ Computer ។ ហើយវាក៏ចាំបាច់តម្រូវឱ្យជ្រើសរើសនូវភាសាកូដ (Language Code) ណាមួយ (ឧទាហរណ៍ C language) សម្រាប់ សរសេរចេញជាកម្មវិធី។

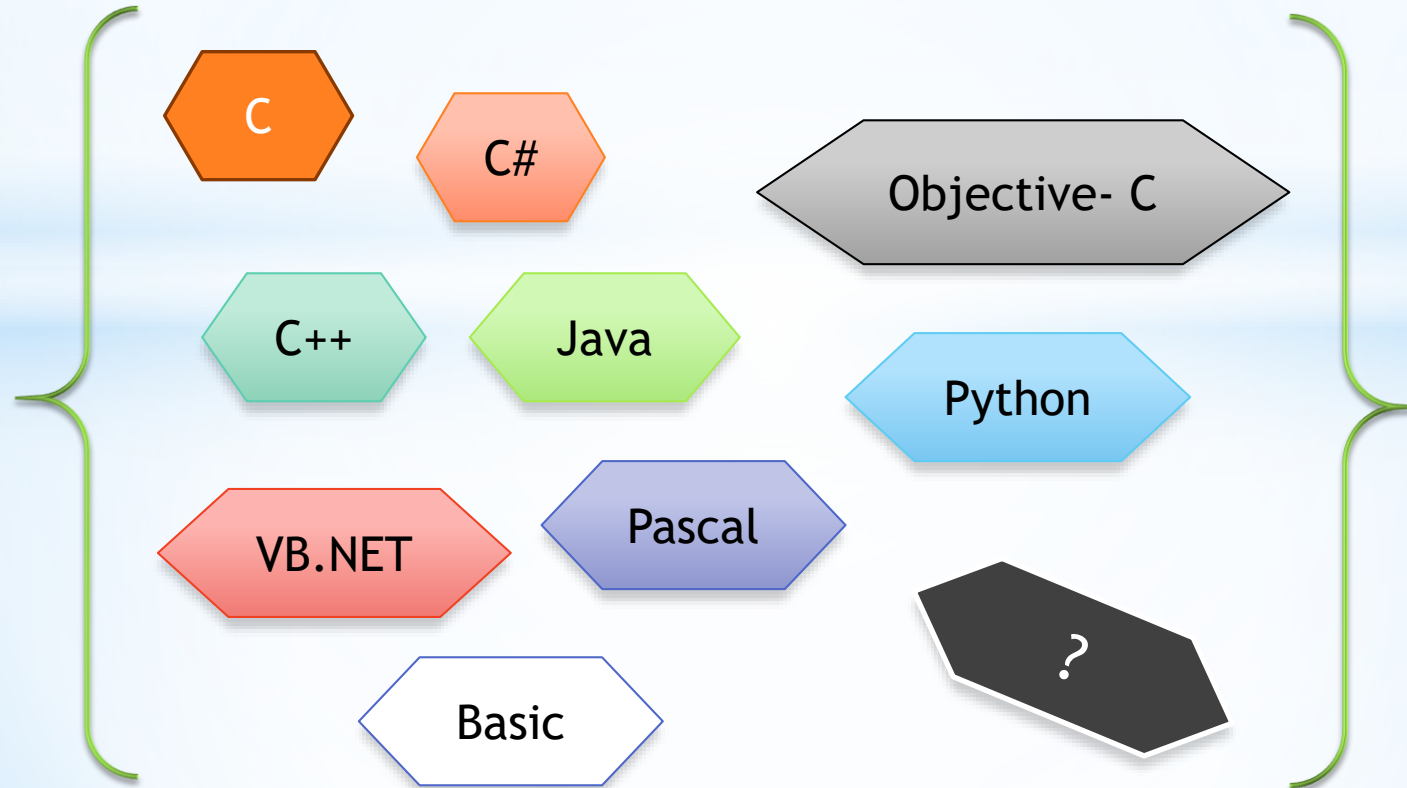
Programming ជាអ្វី?

- ពាក្យថា Programming មិនសំដៅតែលើការសរសេរកម្មវិធី សម្រាប់ Computer តែប៉ុណ្ណោះទេ



C/C++ for beginner

- ភាសាកូដផ្សេងៗសម្រាប់ programming មាន៖



C/C++ for beginner

- ក្នុងចំណោមភាសាកូដ ខាងលើ C ជាភាសា ដំបូងបំផុតដែលមានលក្ខណៈល្អប្រសើរ និង ងាយស្រួលក្នុងការសិក្សា។
- C ជាភាសាប្រើប្រាស់នូវ Compiler ដែលមានប្រសិទ្ធភាពខ្ពស់ជាមួយនឹង Devices
- C ត្រូវបានចាត់ចូលជាពពួក High Level Language ព្រោះវាមានលក្ខណៈ Standard និងមាន structure ច្បាស់លាស់ ដែលផ្តល់ភាពងាយស្រួលដល់អ្នកដើម្បី ឈានទៅសិក្សាភាសា programming ផ្សេងៗទៀតបានយ៉ាងងាយ ដូចជា ៖ C++, C#, Java, Objective C,... ព្រោះភាសាទាំងនេះ សុទ្ធសឹងតែមានមូលដ្ឋានគ្រឹះចេញ ពី ភាសា C ទាំងអស់។

C/C++ for beginner

ប្រវត្តិខ្លះៗពី ភាសា C

ភាសា C ជា High Level Language បង្កើតឡើងដោយលោក Dennis Ritchie ក្នុងកំឡុងឆ្នាំ 1969-1973។ មុនពេលបង្កើតភាសា C គឺមានភាសា Basic Combined Programming Language ដែលសរសេរឡើងដោយលោក Martin Richards នៅឆ្នាំ 1966 និងភាសា B ដែលបង្កើតឡើងនៅឆ្នាំ 1969 ដោយលោក Ken Thompson ។ ដោយយល់ថាភាសាទាំងពីរនៅមានលក្ខណៈមិនទាន់គ្រប់គ្រាន់ លោក Dennis Ritchie ក៏បានបង្កើតគំនិតផ្ទាល់ខ្លួនរបស់គាត់ផ្សំជាមួយនឹងភាសាទាំងពីរដែលមានស្រាប់បង្កើតចេញជាភាសាថ្មីមួយទៀត នោះគឺ C នេះឯង។

C/C++ for beginner

សរសេរកម្មវិធីដោយប្រើភាសា C

- ដើម្បីឱ្យលេចចេញជាកម្មវិធីដោយប្រើភាសា C ត្រូវអនុវត្តតាម 4 ដំណាក់កាល៖

1. Coding/Editing: ជាដំណាក់កាលដំបូង

- សរសេរកូដរបស់កម្មវិធី (Writing program source code)
- គ្រប់គ្រងកម្មវិធី (Managing the programs)
- អភិវឌ្ឍកម្មវិធី (Developing the programs)
- សាកល្បងដំណើរការកម្មវិធី (Testing your programs)

2. Compiling: ជាដំណាក់កាលទីពីរ

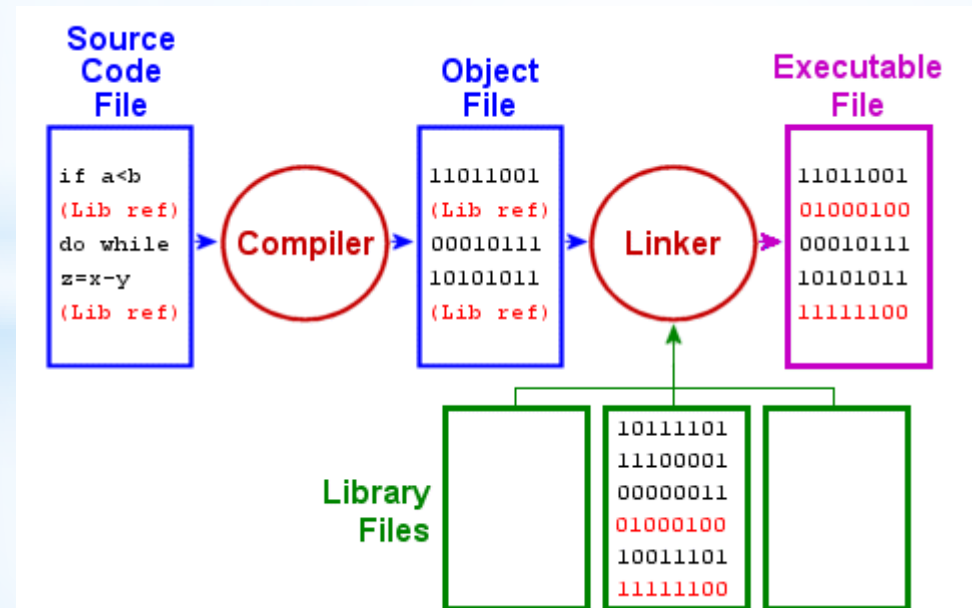
- អង្គបកប្រែ (Compiler) ជាកម្មវិធីមួយដែលមានមុខងារបកប្រែពីភាសាកូដដែល យើងអាចយល់បាន ទៅជាភាសា មួយដែលកុំព្យូទ័រអាចយល់បាន(Binary Language)។
- ពិនិត្យមើលនូវបណ្តាលកំហុសទាំងឡាយដែលកើតមានឡើងនៅពេលសរសេរកូដនៃកម្មវិធី
- ប្រសិនបើការបកប្រែបានសំរេច នោះ Compiler នឹងបង្កើត File មួយដែលមានឈ្មោះដូចគ្នា ជាមួយ File ផ្ទុកកូដដើម តែមាន Extension ជា obj បានន័យថាជា Object File។

Note: Object File គឺជា file ផ្ទុកបណ្តាបញ្ជាដែលបានបកប្រែពីកូដដើម(ផ្ទុកកូដរបស់កម្មវិធីក្រោមទម្រង់ជា Binary Language) តែ file នេះនៅមិនទាន់

អាចដំណើរការបានទេ ដោយរងចាំអង្គតភ្ជាប់ (Linking) ។

សរសេរកម្មវិធីដោយប្រើភាសា C

- ដើម្បីឱ្យលេចចេញជាកម្មវិធីដោយប្រើភាសា C ត្រូវអនុវត្តតាម 4 ដំណាក់កាល៖
 1. Coding/Editing:
 2. Compiling
 3. Linking: អង្គតភ្ជាប់ (Linker) បានតភ្ជាប់បណ្តា file ដែលបំប្លែងដោយ Compiler (បណ្តា Object file) ទៅពិនិត្យជាមួយនឹងបណ្តា file របស់ C library file សំរាប់ផ្គត់ផ្គង់ មើលមានឬក៏អត់។ ឧទាហរណ៍: printf("hello"); //Error អត់មាន printf function ទេ។
 4. Executing: ជាដំណាក់កាលចុងក្រោយដែលអ្នកត្រូវដាក់កម្មវិធីរបស់អ្នកអោយដំណើរការលើ computer ក្រោយពីអ្នកសំរេចបាននូវដំណាក់កាលទាំងបីខាងលើ។



C/C++ for beginner

តើត្រូវសរសេរ code នៅលើអ្វី?

- អ្នកចាំបាច់ត្រូវជ្រើសរើស IDE ណាមួយដែលសមស្របសម្រាប់ភាសា C
- **IDE** ? មកពីពាក្យថា Integrated Development Environment វាជាEditor ឬក៏ជា Tools ដែលប្រើសម្រាប់ការងារ programming
 - Coding
 - Editing
 - Compiling
 - Configuration
 - Testing, ...

C/C++ for beginner

តើភាសា C ត្រូវការ IDE មួយណា?

- **Dev-C++** ជា IDE មួយក្នុងចំណោម IDE ជាច្រើនទៀតដូចជា Turbo C/C++, Borland C++, C-Free, Code:Blocks, XCode...

A screenshot of the Dev-C++ 5.3.0.3 IDE. The window title is "C:\Users\SOLEI\Documents\C++\MyFistApp\FirstApp.c - Dev-C++ 5.3.0.3". The menu bar includes File, Edit, Search, View, Project, Execute, Debug, Tools, CVS, Window, and Help. The toolbar contains various icons for file operations, editing, and execution. The left sidebar shows a project tree with "FirstApp.c" and "main0: int". The main editor area displays the following C code:

```
1 #include<stdio.h>
2 #include<conio.h>
3
4 int main(){
5     printf("This is my first app.");
6     getch();
7 }
```

The line "getch();" is highlighted in light blue. The status bar at the bottom shows "Line: 6 Col: 13 Sel: 0 Lines: 7 Length: 102 Insert Done parsing".

C/C++ for beginner

បង្កើតកម្មវិធីជាឧទាហរណ៍

1. បើក Dev C++
2. បង្កើត Source code file ដោយចូលទៅកាន់ File>New>Source File ឬ ចុច Ctrl+N
3. សរសេរកូដ សម្រាប់ជាឧទាហរណ៍

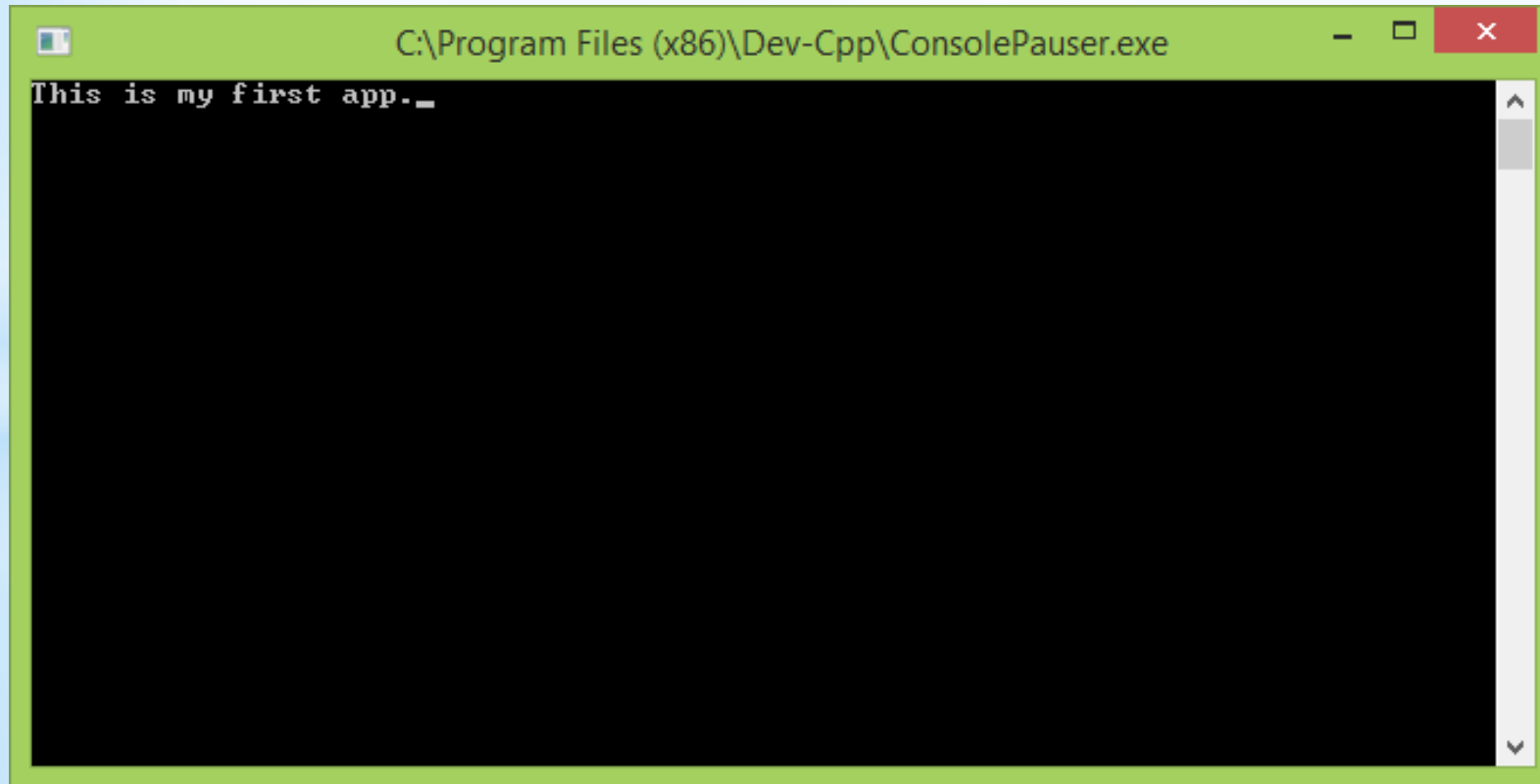
```
#include<stdio.h>
int main() {
    printf("This is my first app.");
    getch();
}
```

C/C++ for beginner

4. ចុច F9 សម្រាប់ Compile រាល់ Source code។
ពេល Compile លើកទីមួយវាទាមទារឱ្យ ជ្រើសរើសទីតាំងសម្រាប់ Save និង
កំណត់ពីប្រភេទ File extension ដែលត្រូវ Compile ដែលជា default វានឹងចាប់
យក C++ source file ដោយស្វ័យប្រវត្ត ដូច្នេះអាចកែទៅជា C Source file វិញ
ដោយ Save file extension ជា .c
5. ចុច F10 ដើម្បីដំណើរការ កម្មវិធីដែលបាន Compile រួចរាល់
ហើយរាល់ការប្រើប្រាស់គឺនៅលើផ្ទាំង Console

C/C++ for beginner

លទ្ធផលបន្ទាប់ពីបាន Compile និង Run



The image shows a screenshot of a Windows console window. The title bar is green and contains the text 'C:\Program Files (x86)\Dev-Cpp\ConsolePauser.exe' along with standard window control buttons (minimize, maximize, close). The main area of the window is black, representing the console output. The text 'This is my first app._' is displayed in white, monospaced font at the top left of the console area. A vertical scrollbar is visible on the right side of the console area.

C/C++ for beginner

Structure of C programming Language

មុននឹងឈានដល់ការសរសេរកូដផ្សេងៗទៀតអ្នកក៏ត្រូវដឹងពី Structure ក៏ដូចជាតម្រូវការដែលត្រូវមានសម្រាប់ C ផងដែរ ។ ចូរមើល Syntax ខាងក្រោម៖

- **#Directives:** សម្រាប់ចាប់ផ្តើមនៃ Program ប្រសិនបើ Program មានប្រើ Function ដែលផ្ទុកក្នុងនោះ។ Directive ចាប់ផ្តើមដោយសញ្ញា # (hash sign)។
- **Global Declaration:** ទិន្នន័យ (Data Type) ឬក៏ Function ()។
- **Main() Function:** ជា Main program ដែលចាំបាច់ត្រូវតែមាន។
- **Open brace:** សម្រាប់ចាប់ផ្តើមនៃ main program ឬក៏ block statements។
- **Statements:** ជាដំណើរការនៅពេល Run Program។
- **Return 0:** សម្រាប់បញ្ចប់ Program ហើយបញ្ជូនព័ត៌មានអោយ Operating System (OS)។
- **Close brace:** សម្រាប់បញ្ចប់ main program ឬក៏ block statements។

*ឧទាហរណ៍:

```
1 #include<stdio.h>
2 //Global declaration
3 int main()
4 {
5     printf("Welcome to C/C++ language");
6     return 0;
7 }
```

Comments

គឺជាការបិទកូដមិនអោយដំណើរការនៅពេល Compile។ Comments ត្រូវបានចែកចេញជាពីរ មានដូចខាងក្រោម:

1. // line comment (use tow slash sign)
2. /* block comment */