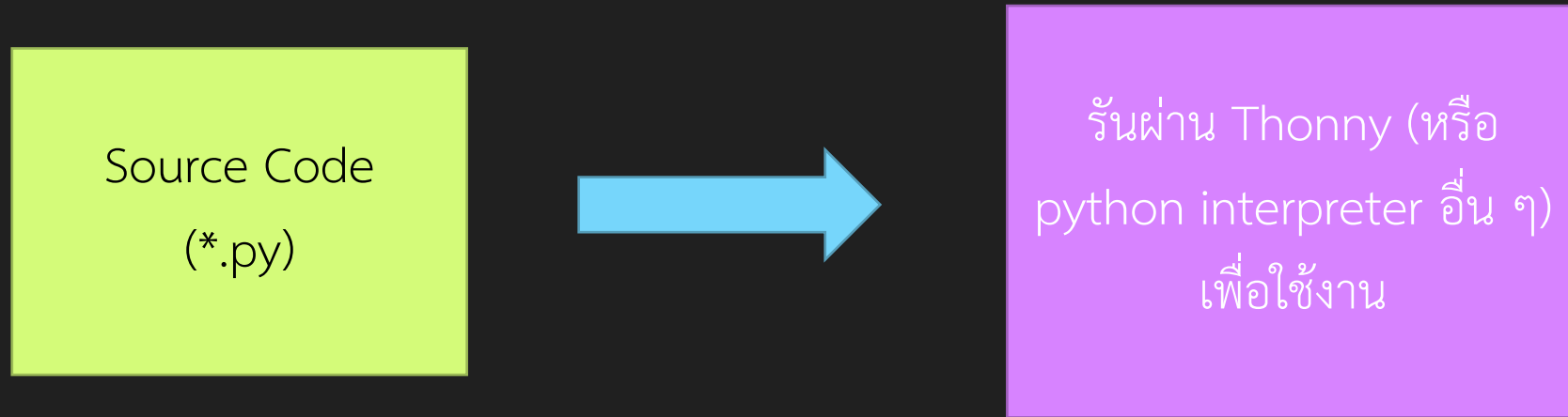


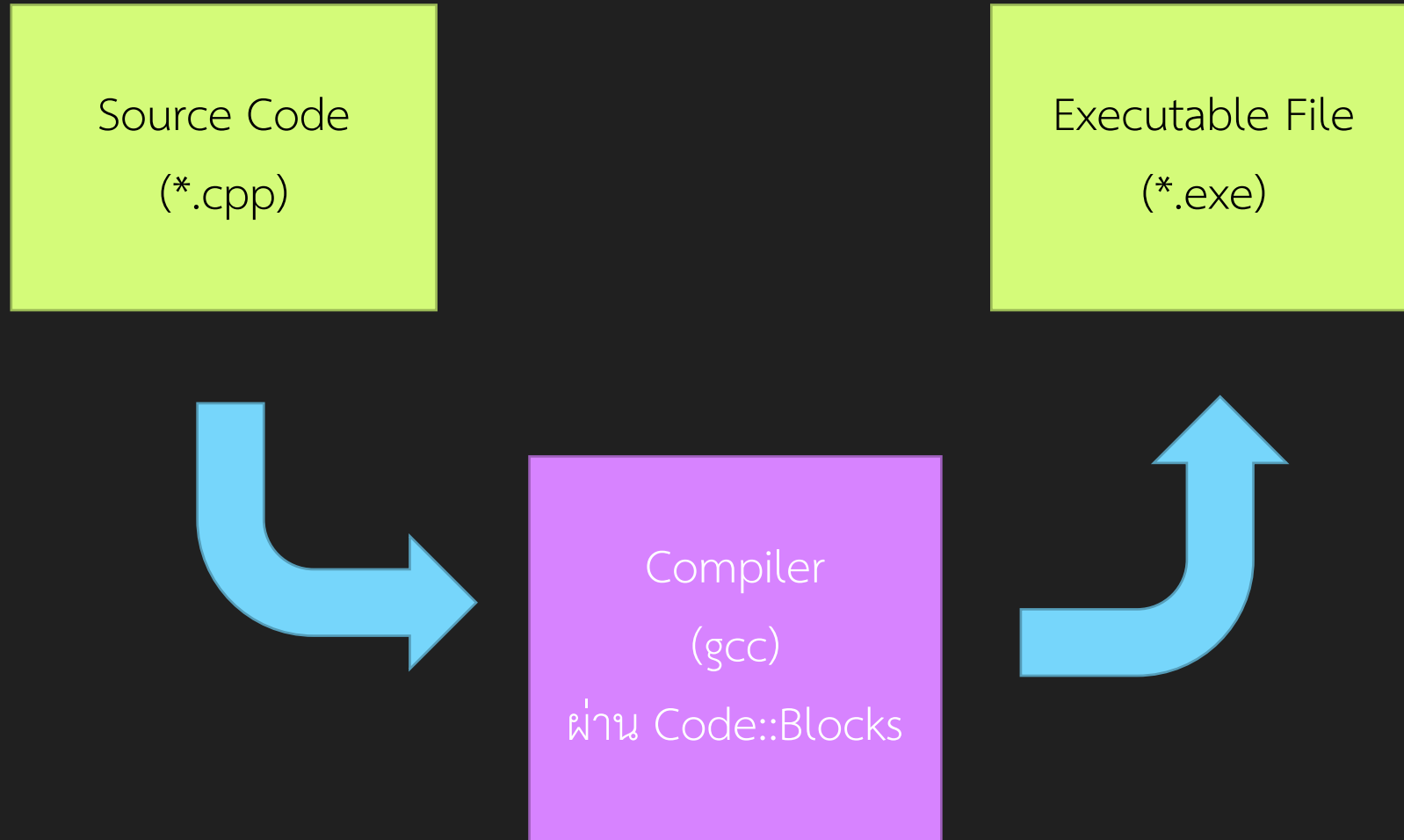
# C++ Programming Language

```
#pragma _CRT_SECURE_NO_WARNINGS
#include "resource.h"
// CDMotionApp
// See DMotion.cpp for the implementation of the class
//
class CDMotionApp : public CWinApp
{
public:
    CDMotionApp();
// Overrides
// ClassWizard generated virtual function overrides
//{{AFX_VIRTUAL(CDMotionApp)
public:
    virtual BOOL InitInstance();
//{{AFX_VIRTUAL
}
```

# Python Work Flow



# C++ Work Flow



# ทดลองเขียน C++

- Download c++ ที่จะใช้ ( [Code::Blocks](#) )
  - C++ มีมาตรฐานอยู่แล้ว เลยมีโปรแกรมสำหรับเขียน C++ หลายตัว
  - วิชา [Data Structure](#) และ [Algorithm Design](#) เลือกใช้ [Code::Blocks](#)
  - ใครอยากจะใช้ตัวอื่นก็ได้ แต่ตอนสอบในศูนย์คอมด้วย grader มีให้ใช้แต่ [Code::Blocks](#)
- ทดลองเขียนโปรแกรม
  - ลองทำโปรแกรมกันสักอันหนึ่ง

# Code::Blocks

- download ได้ที่ [www.codeblocks.org](http://www.codeblocks.org)
  - ให้เลือก download `codeblocks-20.03mingw-setup.exe`
  - อย่าลืมว่าต้องเลือกอันที่มีคำว่า mingw นะครับ!!!

A vintage computer monitor, likely from the late 1980s or early 1990s, is positioned on a grassy field. The monitor's screen displays the text "HELLO WORLD" in a pixelated, monospaced font, arranged in two lines. The monitor has a light-colored bezel and a small, dark rectangular slot at the bottom center. The background consists of a dense line of trees under a bright, overcast sky. The overall scene is captured in a slightly desaturated, blue-toned color palette.

HELLO  
WORLD

# Hello World

## C++

```
#include <iostream>

int main()
{
    std::cout << "Hello, CP!" << std::endl;
    return 0;
}
```

## JAVA

```
import java.io.*;

class HelloWorldApp
{
    public static void main(String[] args)
    {
        System.out.println("Hello CP!");
    }
}
```

## Python

```
print("Hello CP!")
```

# เรื่องหลัก ๆ ที่ C++ ต่างจาก Java

- Syntax แทบจะเหมือนกัน
- function ไม่จำเป็นต้องอยู่ใน class
- ตัวแปรโดยปรกติคือ “พื้นที่ในหน่วยความจำที่ไว้เก็บข้อมูล”
  - เหมือน POD ใน java
- new แล้วต้อง delete!!!
- ยิบย่อย
  - Input / Output ต่างกัน
  - Main function ต่างกัน



# เรื่องหลัก ๆ ที่ C++ ต่างจาก Python

- ตัวแปรต้องมีการ “ประกาศ” ก่อนใช้งาน
- ตัวแปรต้องมีการระบุ “ประเภท”
- ต้องมี “ปีกกา” ล้อมรอบ block ของ code (ไม่ได้ใช้การย่อหน้า)
- syntax ต่างกัน

# เอกสารแนะนำให้ลองอ่าน สำหรับคนที่ใช้ Python

- แบบสั้น ๆ
  - <http://watts.cs.sonoma.edu/UsefulStuff/C++ForPythonProgrammers.pdf>
    - จาก Dr. Tia Watts, Sonoma State University
- แบบยาว ๆ
  - <https://runestone.academy/runestone/books/published/cpp4python/index.html>
    - โดย Jan Pearce, Berea College, and Brad Miller, Runestone
    - มีแบบฝึกหัด

# C++ Tutorial

- <http://www.cplusplus.com/doc/tutorial/>
  - ควรอ่านและลองทำถึงหัวข้อ Class (II)
    - ไม่ต้องรีบ ค่อย ๆ ทำไปก็ได้ ยังไม่ได้ใช้หมดทุกอย่างในนี้ ณ วันนี้

# ตัวแปร

- ต้องประกาศ
- ต้องระบุประเภท
- ตัวแปร โดยปรกติเป็น “กล่อง” ที่เก็บข้อมูล ไม่ใช่ของที่ “อ้างอิง” ข้อมูล

```
#include <iostream>

using namespace std;

int main() {
    int x;
    x = 10;
    bool y = true;
    cout << y << endl;
    cout << (x+20) << endl;
    x = y // ไม่ได้
}
```

# If statement

- ต้องมีวงเล็บ ตรงเงื่อนไข
- ใช้ { } เป็นตัวระบุ suite (ใน c++ เรียก block)
- ไม่มี elif ต้องใช้ else if

```
int main() {  
    int age;  
    cout << "Please enter your age:";  
    cin >> age;  
    if ( age < 5 ) {  
        cout << "You are a kid!\n";  
    } else if ( age < 100 ) {  
        cout << "You are not old!\n";  
    } else {  
        cout << "You live long!\n";  
    }  
    return 0;  
}
```

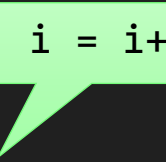
# For loop

- เหมือน java
- ประกอบด้วย 3 ส่วน
  - initial
  - condition
  - iteration

```
#include <iostream>

using namespace std;

int main()
{
    for (int i = 0; i < 10; i++) {
        cout << "i = " << i << endl;
    }
    return 0;
}
```



# While loop

```
int main()
{
    int x = 0;
    while (x < 10 ) {
        cout << "x = " << x << endl;
        x++;
    }
    return 0;
}
```

```
int main() {
    int x = 20;
    do {
        x--;
        cout << x << endl;
    } while (x > 10)
}
```

# Function

- Pass by value vs pass by reference

```
void pass_by_value(int x) {  
    cout << "X is" << x << endl;  
    x = 30;  
}  
  
void pass_by_reference(int &x) {  
    cout << "X is" << x << endl;  
    x = 40;  
}
```

```
int main() {  
    cout << "Pass by Value, direct" << endl;  
    pass_by_value(10);  
    cout << endl;  
  
    int x = 20;  
    cout << "Pass by value, variable" << endl;  
    pass_by_value(x);  
    cout << "outside PbR function x = " << x << endl;  
    cout << endl;  
  
    cout << "Pass by reference" << endl;  
    pass_by_reference(x);  
    cout << "outside PbR function x = " << x << endl;  
  
    //the following line cannot be compiled  
    //because we need reference  
    //pass_by_reference(20);  
}
```