

LECTURE 01

INTRODUCTION



pythonTM

Big-O Coding

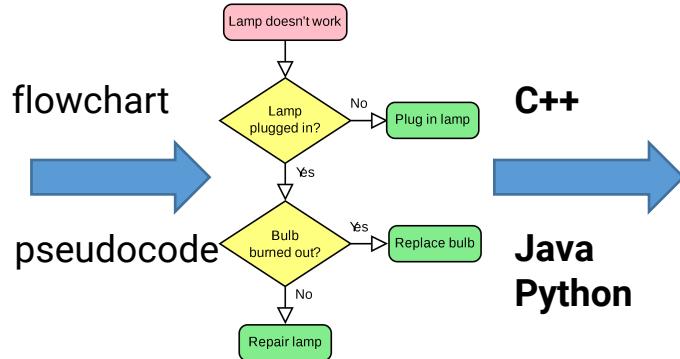
Website: www.bigocoding.com

Chương trình (program, application)

- Một dãy các chỉ thị (instruction).
- Điều khiển sự hoạt động của máy tính.
- Hỗ trợ con người giải quyết một công việc nào đó.

Lập trình (programming)

- Quá trình tạo ra 1 chương trình.



Problem

Algorithm

C++
Java
Python

```
def main:
    lamp = 0
    if lamp == 1:
        plugin(lamp)
    else:
        replace(lamp)
```

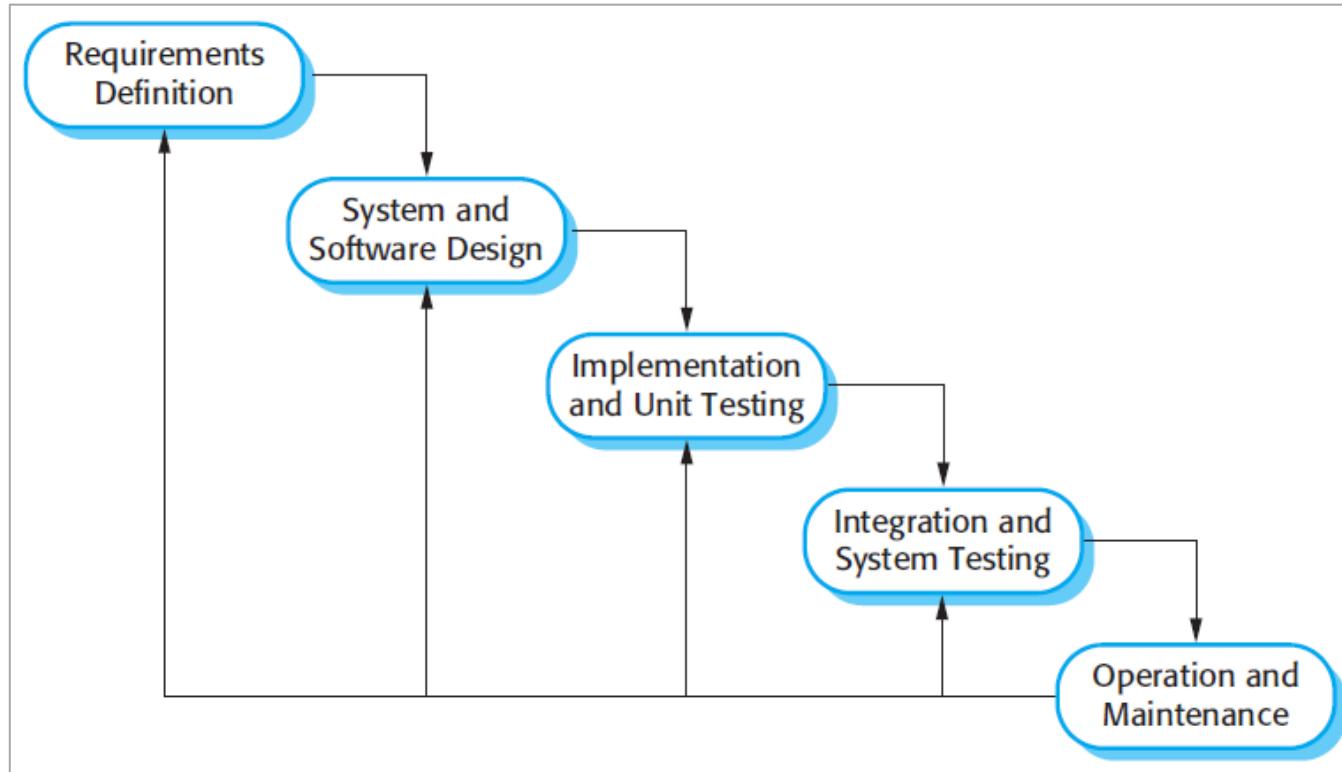
main()

Compiler
Interpreter



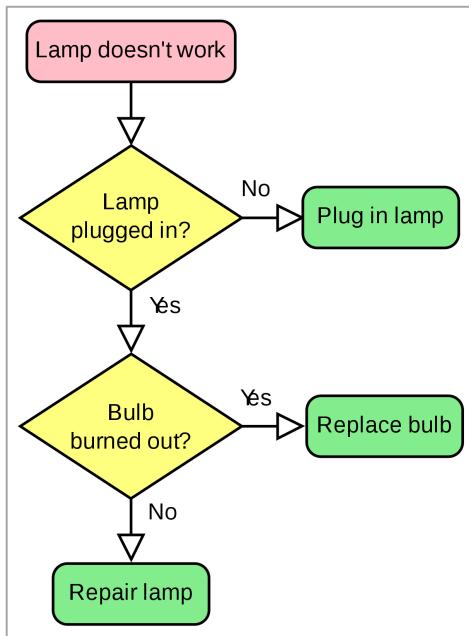
Program

Qui trình phần mềm (software process)



2 cách trình bày giải thuật (algorithm)

Flowchart

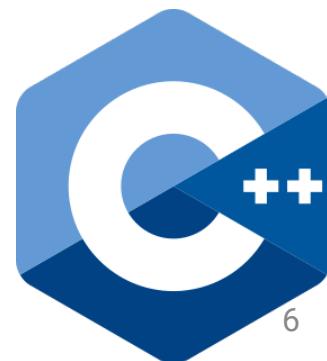


Pseudo-code

```
if lamp pluggin in then  
    if Bulb burned out then  
        Repair the bulb  
    else  
        Replace the lamp  
    else  
        Plugin in the lamp
```

Giới thiệu C++

- Bjarne Stroustrup, 1985.
- C++ 17.
- Mọi lập trình viên chuyên nghiệp đều từng trải nghiêm qua.
- System programming, embedded, resource-constrained and large systems.



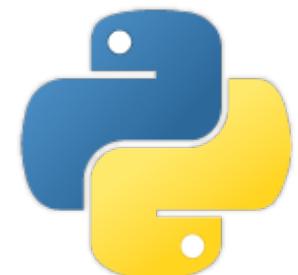
Giới thiệu Java

- James Gosling, 1995.
- Java 10 (20/03/2018).
- Ưa chuộng tại nhiều trường ĐH ở VN và thế giới.
- Mobile programming, web development.



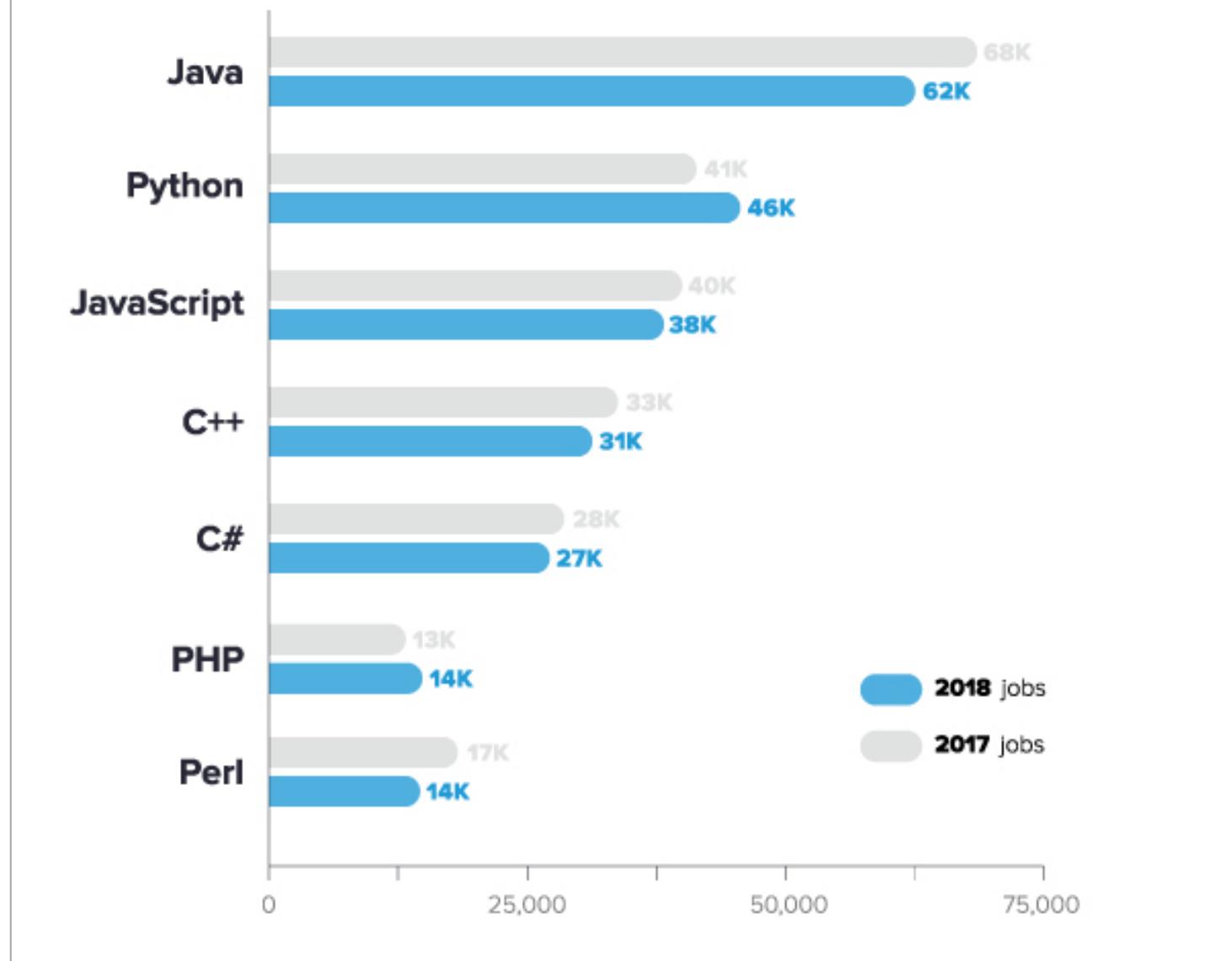
Giới thiệu Python

- Guido van Rossum, 1991.
- 3.7.0 (27/06/2018).
- Thích hợp cho người mới bắt đầu (newbie).
- Scientific and numeric computing, web development.



Job postings containing top languages

Indeed.com - November, 17th 2017

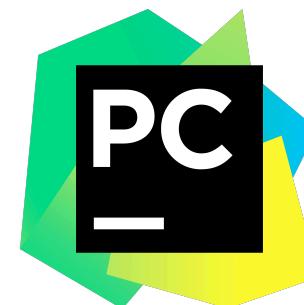
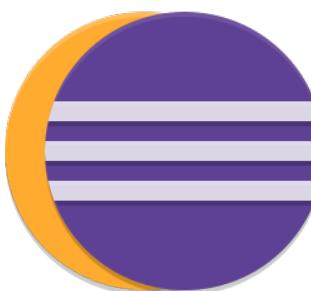
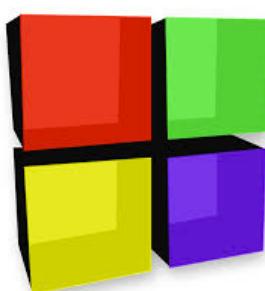


Mã nguồn (source code)

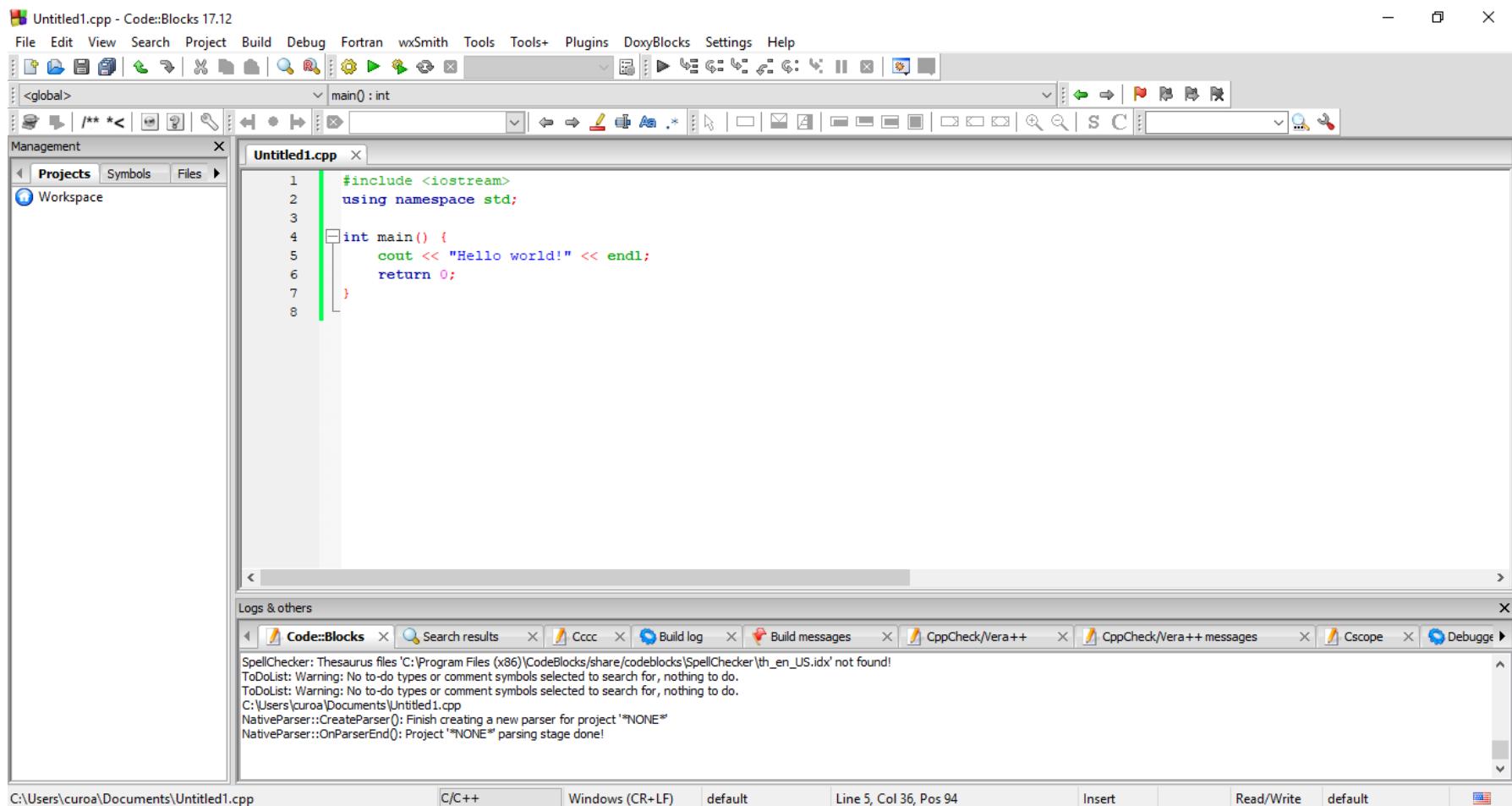
- Mã nguồn (source code) là một tập tin văn bản (text file).
 - C++: helloworld.**cpp**
 - Java: helloworld.**java**
 - Python: helloworld.**py**
- Ta có thể sử dụng bất kì text editor đơn giản nào trên máy để soạn thảo.
 - Notepad
 - TextEdit

Giới thiệu IDE

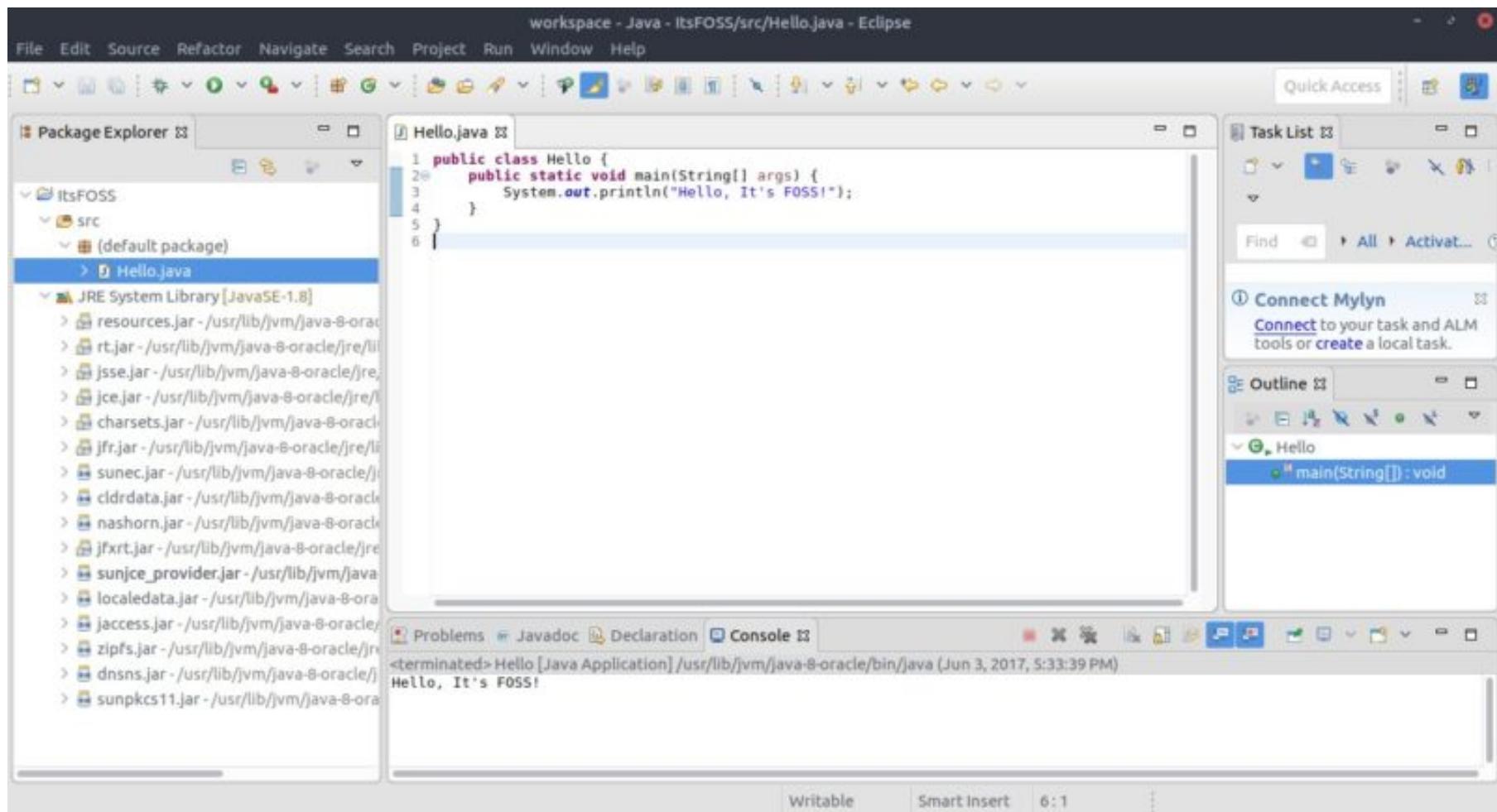
- Lập trình viên (developer) thường sử dụng các IDE (Integrated Development Environment) vì:
 - Hỗ trợ biên soạn source code. (**EDIT**)
 - Hỗ trợ biên dịch source code. (**COMPILE**)
 - Hỗ trợ chạy thử, xem kết quả. (**RUN**)
 - Hỗ trợ tìm lỗi. (**DEBUG**)



CodeBlocks



Eclipse



PyCharm

The screenshot shows the PyCharm Professional 2016.2.3 interface. The title bar indicates the project is 'angry-birds-python' and the file being edited is 'main.py'. The menu bar includes File, Edit, View, Navigate, Code, Refactor, Run, Tools, VCS, Window, and Help. The toolbar has icons for Run, Stop, Step, and others. The Project tool window on the left shows the project structure with 'src' containing 'main.py', 'level.py', 'polygon.py', and 'characters.py', along with 'resources' and 'pymunk-4.0.0'. The main editor window displays the 'main.py' code:

```
1 import ...
2 current_path = os.getcwd()
3 sys.path.insert(0, os.path.join(current_path, "../pymunk-4.0.0"))
4 import pymunk as pm
5 from characters import Bird
6 from level import Level
7
8 pygame.init()
9 screen = pygame.display.set_mode((1200, 650))
10 redbird = pygame.image.load(
11     "../resources/images/red-bird3.png").convert_alpha()
12 background2 = pygame.image.load(
13     "../resources/images/background3.png").convert_alpha()
14 sling_image = pygame.image.load(
15     "../resources/images/sling-3.png").convert_alpha()
16 full_sprite = pygame.image.load(
17     "../resources/images/full-sprite.png").convert_alpha()
18 rect = pygame.Rect(181, 1050, 50, 50)
19 cropped = full_sprite.subsurface(rect).copy()
20 pig_image = pygame.transform.scale(cropped, (30, 30))
21 buttons = pygame.image.load(
22     "../resources/images/selected-buttons.png").convert_alpha()
23 pig_happy = pygame.image.load(
24     "../resources/images/pig_failed.png").convert_alpha()
25 stars = pygame.image.load(
26     "../resources/images/stars-edited.png").convert_alpha()
27 rect = pygame.Rect(0, 0, 200, 200)
28 star1 = stars.subsurface(rect).copy()
29 rect = pygame.Rect(204, 0, 200, 200)
30 star2 = stars.subsurface(rect).copy()
31 rect = pygame.Rect(426, 0, 200, 200)
32 star3 = stars.subsurface(rect).copy()
33 rect = pygame.Rect(164, 10, 60, 60)
34 pause_button = buttons.subsurface(rect).copy()
35 rect = pygame.Rect(24, 4, 100, 100)
36 replay_button = buttons.subsurface(rect).copy()
37 rect = ovname.Rect(142, 365, 130, 100)
```

A 'Tool Windows Quick Access' tooltip is visible in the bottom-left corner, and a 'Got it!' button is present. The status bar at the bottom shows 12 items (6.9 MB), Free space: 502.1 GB, 2 processes running..., LF+, UTF-8+, Git: master, and the current time 9:50:47 pm.

Một chương trình C++ đơn giản

```
// L01P01.cpp
#include <iostream>
using namespace std;

int main()
{
    cout << "Hello, World!!!";
    return 0;
}
```

thư viện (library) nhập xuất

1 câu lệnh (statement) kết thúc bằng dấu ;

Mỗi chương trình bắt đầu bằng hàm main()

Một chương trình Java đơn giản

```
// L01P01.java
public class L01P01 {
    public static void main(String[] args) {
        System.out.print("Hello, World!!!");
    }
}
```

kết thúc câu lệnh bằng dấu ; (như C++)

mỗi chương trình bắt đầu bằng hàm main()

tên class và tên file phải giống nhau

Một chương trình Python đơn giản

```
# L01P01.py  
print("Hello, World!!!")
```

không cần include thư viện khi nhập xuất

không cần hàm main

không cần dấu ; cuối câu lệnh

In dữ liệu ra màn hình



- Sử dụng toán tử <<.

```
cout << "Hello, World!!!";
```



- Sử dụng hàm System.out.print() / println()

```
System.out.print("Hello, World!!!");
```



- Sử dụng hàm print()

```
print("Hello, World!!!")
```

Kiểu dữ liệu chuỗi

- “Hello, World!!!” trong ví dụ ở slide trước được gọi là một **chuỗi** (string), đặt trong cặp dấu nháy đôi (double quotes).
- Chuỗi là một trong những kiểu dữ liệu quan trọng trong các ngôn ngữ lập trình.
- Chuỗi dùng để lưu một câu, lưu một đoạn văn, lưu tên học viên, lưu địa chỉ, lưu tên sản phẩm...
- Ta sẽ học chi tiết về chuỗi ở Buổi 8.

BT1 – HELLO, WORLD!!!

- Bài tập in ra dòng chữ “Hello, World!!!” là một bài tập cơ bản cho mọi người mới bắt đầu học ngôn ngữ lập trình. Bạn hãy in ra “Hello, World!!!” ra màn hình (không có dấu “ ”).



Kiểu dữ liệu số (Number)

- Có 2 kiểu dữ liệu số được dùng phổ biến: integer và floating point number.
- Integer: số nguyên, bao gồm: các số âm, số 0 và các số dương.
 - VD: 1, 2, -5, 1000.
- Floating point number: số thực.
 - VD: 1.2, -0.5, 2e2 (200), 4E3 (4000).

VD: in số ra màn hình

```
cout << 3;  
cout << 3.14;  
cout << 4E3;
```



```
System.out.print(3);  
System.out.print(3.14);  
System.out.print(4E3);
```



```
print(3)  
print(3.14)  
print(4E3)
```



Các phép toán số học (operator) trong C++

```
cout << (9 + 2);
cout << (9 - 2);
cout << (9 * 2);
cout << (1.0 * 9 / 2);

cout << (9 % 2);
cout << (9 / 2);

cout << (10 * (2 + 3) - 3);
```



Các phép toán số học (operator) trong Java

```
System.out.print(9 + 2);  
System.out.print(9 - 2);  
System.out.print(9 * 2);  
System.out.print(1.0 * 9 / 2);
```

```
System.out.print(9 % 2);  
System.out.print(9.0 % 2.0);  
System.out.print(9 / 2);
```

```
System.out.print(10 * (2 + 3) - 3);
```



Các phép toán số học (operator) trong Python

```
print(9 + 2)
```

```
print(9 - 2)
```

```
print(9 * 2)
```

```
print(9 / 2)
```

```
print(9 % 2)
```

```
print(9.0 % 2.0)
```

```
print(9 ** 2)
```

```
print(9 // 2)
```

```
print(9.0 // 2.0)
```

```
print(10 * (2 + 3) - 3)
```



Biến

```
int a;  
a = 9 + 2;  
cout << a;
```



```
int a;  
a = 9 + 2;  
System.out.print(a);
```



```
# Không cần khai báo  
a = 9 + 2  
print(a)
```



- a được gọi là một biến (variable).
- Biến dùng để lưu một kết quả tính toán.
- **Mỗi biến chiếm 1 vùng nhớ trong bộ nhớ (memory, RAM).**

Kiểu dữ liệu số trong C++

- Kiểu số nguyên

Kiểu dữ liệu	Độ lớn (bytes)	Miền giá trị
short	2	-32.768 ... 32.767
int	4	-2.147.483.648 ... 2.147.483.647
long	4	-2.147.483.648 ... 2.147.483.647
long long	8	-9.223.372.036.854.775.808 ... 9.223.372.036.854.775.807

- Kiểu số thực

Kiểu dữ liệu	Độ lớn (bytes)	Miền giá trị
float	4	$1.40239846 \times 10^{-45}$... $3.40282347 \times 10^{38}$
double	8	$4.9406564584124654 \times 10^{-324}$... $1.7976931348623157 \times 10^{308}$

Kiểu dữ liệu số trong Java

- Kiểu số nguyên

Kiểu dữ liệu	Độ lớn (byte)	Miền giá trị
byte	1	-128 ... 127
short	2	-32.768 ... 32.767
int	4	-2.147.483.648 ... 2.147.483.647
long	8	-9.223.372.036.854.775.808 ... 9.223.372.036.854.775.807

- Kiểu số thực

Kiểu dữ liệu	Độ lớn (bytes)	Miền giá trị
float	4	$1.40239846 \times 10^{-45}$... $3.40282347 \times 10^{38}$
double	8	$4.9406564584124654 \times 10^{-324}$... $1.7976931348623157 \times 10^{308}$

Kiểu dữ liệu số trong Python

- Python không khai báo biến.
- Kiểu dữ liệu của 1 biến khi bạn gán giá trị cho biến đó.
- Kiểu dữ liệu integer và floating point number trong Python không giới hạn độ lớn của giá trị.

Qui tắc đặt tên biến (naming convention)

- Chỉ sử dụng các chữ cái (character), các số (digit) và dấu underscore.
- Không bắt đầu bằng số.
- Không dùng khoảng trắng (whitespace) hay các kí tự đặc biệt.
- Thường đặt tên biến theo kiểu camelCase (**myAge**) hoặc lower_case (**my_age**).
- C++, Java và Python là các ngôn ngữ lập trình có phân biệt chữ hoa chữ thường (case sensitive).

Ví dụ đặt tên biến và sử dụng đúng

```
int thisYear;  
thisYear = 2018;  
int yearOfBirth;  
yearOfBirth = 1998;  
int myAge;  
myAge = thisYear - yearOfBirth;  
cout << myAge;
```



```
thisYear = 2018  
  
yearOfBirth = 1998  
  
myAge = thisYear - yearOfBirth  
print(myAge)
```



```
int thisYear;  
thisYear = 2018;  
int yearOfBirth;  
yearOfBirth = 1998;  
int myAge;  
myAge = thisYear - yearOfBirth;  
System.out.print(myAge);
```



Ví dụ đặt tên biến và sử dụng sai!!!

```
int my age = 3;  
bool 1styear = True;  
int this-year = 2017;  
cout << thisYear;  
ans = 31;
```



```
my age = 3  
1styear = True  
this-year = 2017  
print(thisYear)
```



```
int my age = 3;  
bool 1styear = True;  
int this-year = 2018;  
System.out.print(thisYear);  
ans = 31;
```



Một số toán tử số học khác

- $a = b$, lấy kết quả tính toán bên phải (b) gán vào biến a .
- $a += b$, tương đương với $a = a + b$
- $a -= b$, tương đương với $a = a - b$
- $a *= b$, tương đương với $a = a * b$
- $a /= b$, tương đương với $a = a / b$
- $a %= b$, tương đương với $a = a \% b$

Một số toán tử số học khác

- $a++$, tương đương với $a = a + 1$ (C++, Java)
- $a --$, tương đương với $a = a - 1$ (C++, Java)
- $a **= b$, tương đương với $a = a ** b$ (Python)
- $a //= b$, tương đương với $a = a // b$ (Python)

Ví dụ

```
int a;  
a = 10;  
a += 2;  
cout << a;  
a -= 2;  
cout << a;  
a *= 2;  
cout << a;  
a /= 2;  
cout << a;
```

// C++ không có toán tử **

// C++ không có toán tử //

```
a %= 2;  
cout << a;
```



Không cần khai báo

```
a = 10  
a += 2  
print(a)  
a -= 2  
print(a)  
a *= 2  
print(a)  
a /= 2  
print(a)
```

```
a **= 2  
print(a)  
a //= 2  
print(a)  
a %= 2  
print(a)
```



Ví dụ

```
int a;
a = 10;
a += 2;
System.out.print(a);
a -= 2;
System.out.print(a);
a *= 2;
System.out.print(a);
a /= 2;
System.out.print(a);

// Java không có toán tử **

// Java không có toán tử //

a %= 2;
System.out.print(a);
```



In nhiều nội dung cùng lúc

- Đặt yêu cầu:
 - $a = 5, b = 7, c = a + b = 12.$
 - Cần in ra theo định dạng $5 + 7 = 12.$

In nhiều nội dung cùng lúc (C++)

- C++: sử dụng liên tiếp các toán tử <<.

```
int a = 5;  
int b = 7;  
int c = a + b;  
cout << a << " + " << b << " = " << c;
```



In nhiều nội dung cùng lúc (Java)

- Java: sử dụng hàm printf.
 - %d: số nguyên (format specifier)
 - %f: số thực

```
int a = 5;
int b = 7;
int c = a + b;
System.out.printf("%d + %d = %d", a, b, c);
```

In nhiều nội dung cùng lúc (Python)

- Python: sử dụng hàm format().

```
a = 5  
b = 7  
c = a + b  
print('{0} + {1} = {2}'.format(a, b, c))
```



In số thực

```
#include<iomanip>
using namespace std;

cout << fixed << setprecision(3) << 3.141592653589793238462;
```



```
System.out.printf("The value of PI is approximately %.3f",
3.141592653589793238462);
```



```
print('The value of PI is approximately
{0:.3f}'.format(3.141592653589793238462))
```



Đọc 2 số nguyên trên 2 dòng khác nhau

```
7  
8
```

```
#include <iostream>  
using namespace std;  
  
int a, b;  
cin >> a >> b;
```



```
import java.util.Scanner;  
  
int a, b;  
Scanner sc = new Scanner(System.in);  
a = sc.nextInt();  
b = sc.nextInt();
```



```
a = int(input())  
b = int(input())
```



Đọc 2 số thực trên 2 dòng khác nhau

7.1

8.2

```
#include <iostream>
using namespace std;

double a, b;
cin >> a >> b;
```



```
import java.util.Scanner;
```

```
double a, b;
Scanner sc = new Scanner(System.in);
a = sc.nextDouble();
b = sc.nextDouble();
```



```
a = float(input())
b = float(input())
```



Đọc 2 số nguyên trên cùng 1 dòng

```
7 8
```

```
#include <iostream>
using namespace std;

int a, b;
cin >> a >> b;
```



```
import java.util.Scanner;

int a, b;
Scanner sc = new Scanner(System.in);
a = sc.nextInt();
b = sc.nextInt();
```



```
a,b = map(int, input().split())
```



Đọc 2 số thực trên cùng 1 dòng

7.1 8.2

```
#include <iostream>
using namespace std;

double a, b;
cin >> a >> b;
```



```
import java.util.Scanner;

double a, b;
Scanner sc = new Scanner(System.in);
a = sc.nextDouble();
b = sc.nextDouble();
```



```
a,b = map(float, input().split())
```



BT2 – TÍNH TỔNG

- Viết chương trình cho phép người dùng nhập vào 2 số nguyên lần lượt là số kẹo của Upan và Ipan.
- Bạn hãy tính tổng số kẹo của 2 bạn ấy và in kết quả ra màn hình.



BT3 – TÍNH TUỔI

- Nhập vào năm sinh của một người. Tính tuổi của người đó tính đến năm 2018.



BT4 – TỔNG ĐIỂM THI

- Viết chương trình nhập vào điểm 3 môn Toán, Khoa học tự nhiên, Anh Văn. Tính tổng điểm thi 3 môn của thí sinh đó.



BT5 – ĐẾM NÚT

- Hôm nay là ngày nghỉ, Ben cùng chơi một trò chơi với các bạn của mình.
- Luật trò chơi rất đơn giản, mỗi người sẽ nhận được một con số nguyên dương gồm có 5 chữ số.
- Nếu ai có số nút từ con số này là lớn nhất thì chiến thắng.
- Hãy giúp Ben tính xem số nút của con số mình nhận được là bao nhiêu.



Hỏi đáp



BT1 – Gợi ý

- Sử dụng lệnh xuất để in chữ “Hello, World!!!”
- Nhớ type chính xác từng kí tự, vì máy chấm BigO kiểm tra rất gắt gao.



BT2 – Gợi ý

1. Sử dụng lệnh nhập để đọc dữ liệu vào 2 biến a, b.
 - C++: nên sử dụng kiểu dữ liệu long long.
 - Java: nên sử dụng kiểu dữ liệu long.
 - Python: lưu ý 2 số trên cùng 1 dòng.
2. Cộng kết quả vào c.
3. Sử dụng lệnh xuất để in kết quả ra màn hình.



BT3 – Gợi ý

1. Sử dụng lệnh nhập, để đọc vào năm sinh.
2. Tính tuổi, lấy mốc năm 2018.
3. Sử dụng lệnh xuất, in kết quả ra màn hình.



BT4 – Gợi ý

1. Sử dụng lệnh nhập, để đọc vào điểm 3 môn toán, khtn và văn.
 - Lưu ý: số thực.
2. Tính điểm tổng.
3. Sử dụng lệnh xuất, in kết quả ra màn hình.
 - Lưu ý: in 1 chữ số thập phân.



BT5 – Gợi ý

1. Sử dụng lệnh nhập, để đọc vào số nguyên n.
2. Sử dụng toán tử %, để tách ra 5 chữ số, lưu vào 5 biến digit1, digit2, digit3, digit4 và digit5 như sau:
 - a. $\text{digit1} = n \% 10 \rightarrow n = n / 10$
 - b. $\text{digit2} = n \% 10 \rightarrow n = n / 10$
 - c. ...
3. Cộng giá trị 5 biến digit lại và % 10 lần nữa.
4. Sử dụng lệnh xuất, in kết quả ra màn hình.



Chạy source code C++ từ commandline

- Trình biên dịch (compiler) C++ biên dịch từ file .cpp thành file .o.
- Sau đó, compiler chuyển file .o thành file executable.

```
g++ -o L01P01.exe L01P01.cpp
```

- Chạy chương trình

```
L01P01.exe
```

Chạy source code Java từ command line

- Trình thông dịch (interpreter) biên dịch từ file .java thành file .class.

```
javac L01P01.java
```

- Trên máy tính cần có máy ảo Java (JVM, Java Virtual Machine) để chạy file .class này.

```
java L01P01
```

Chạy source code Python từ command line

- Chạy trực tiếp từ source code Python.

```
python L01P01.py
```

Terminology

Program	Application
Programming	Instruction
Algorithm	Source code
Compiler	Interpreter
Software process	Text editor
IDE	Developer
Library	Operator
Data type	Integer
Floating point number	String
Variable	Memory, RAM
Memory block	Character
Digit	Underscore
Whitespace	Case sensitive
Case insensitive	Naming convention