

Thuật toán và cách biểu diễn thuật toán

Khoá học: Python căn bản

Kiểm tra bài trước

Hỏi và trao đổi về các khó khăn gặp phải trong bài “Nhập môn Python”

Tóm tắt lại các phần đã học từ bài “Nhập môn Python”

Mục tiêu

- Trình bày được ý nghĩa của lập trình và ngôn ngữ lập trình
- Trình bày được khái niệm thuật toán
- Mô tả được giải thuật sử dụng được pseudo-code và Flowchart
- Hướng dẫn sử dụng Jupyter Notebook

Thảo luận

Lập trình và ngôn ngữ lập trình

Lập trình và Ngôn ngữ lập trình

- **Lập trình** là quá trình tạo ra tập các chỉ dẫn (instruction) để ra lệnh cho máy tính hoàn thành một công việc (task) nào đó
- Lập trình bao gồm rất nhiều hoạt động: Tìm hiểu yêu cầu, phân tích, thiết kế, viết code, kiểm thử, triển khai, bảo trì, mở rộng...
- **Ngôn ngữ lập trình** là phương tiện để lập trình viên viết ra các chỉ dẫn cho máy tính

Các loại ngôn ngữ lập trình

- Có nhiều loại ngôn ngữ lập trình khác nhau, phục vụ cho các mục đích khác nhau
- Chẳng hạn, chúng ta có các ngôn ngữ lập trình web, ngôn ngữ lập trình desktop, ngôn ngữ lập trình mobile...
- Học lập trình có nghĩa là:
 - Học tư duy giải quyết vấn đề
 - Học một (hoặc một số) ngôn ngữ lập trình
- Học một ngôn ngữ lập trình có nghĩa là:
 - Học tư duy của ngôn ngữ đó
 - Học cú pháp của ngôn ngữ đó

Học lập trình

- Học lập trình có nghĩa là:
 - Học tư duy giải quyết vấn đề
 - Học một (hoặc một số) ngôn ngữ lập trình
- Học một ngôn ngữ lập trình có nghĩa là:
 - Học tư duy của ngôn ngữ đó
 - Học cú pháp của ngôn ngữ đó

Thảo luận

Khái niệm về thuật toán

Thuật toán (Algorithm)

- Thuật toán/giải thuật bao gồm các chỉ thị để giải quyết một vấn đề
- Có thể sử dụng các cách khác nhau để mô tả thuật toán, các cách thông dụng là:
 - Mã giả (pseudo-code)
 - Lưu đồ (flowchart)

Thuật toán: Ví dụ

- Bài toán:
 - Có hai bình A và B đựng hai loại chất lỏng khác nhau, chẳng hạn bình A đựng rượu, bình B đựng nước mắm. Yêu cầu hoán đổi (swap) chất lỏng đựng trong hai bình
- Thuật toán:
 - Yêu cầu phải có thêm một bình thứ ba gọi là bình C
 - Bước 1: Đổ rượu từ bình A sang bình C
 - Bước 2: Đổ nước mắm từ bình B sang bình A
 - Bước 3: Đổ rượu từ bình C sang bình B

Các cách biểu diễn giải thuật thông dụng



- Mã giả (Pseudo-code)
- Lưu đồ (Flowchart)
- Ngôn ngữ lập trình

Mã giả

- Liệt kê tuần tự các bước bằng ngôn ngữ tự nhiên để biểu diễn thuật toán
- Ưu điểm
 - Đơn giản, không cần kiến thức về cách biểu diễn (lưu đồ, ngôn ngữ lập trình)
- Nhược điểm
 - Dài dòng, không cấu trúc
 - Đôi lúc khó hiểu, không diễn đạt được thuật toán







Mã giả: Ví dụ

- Giải thuật giải phương trình bậc nhất dạng $ax + b = 0$ như sau:

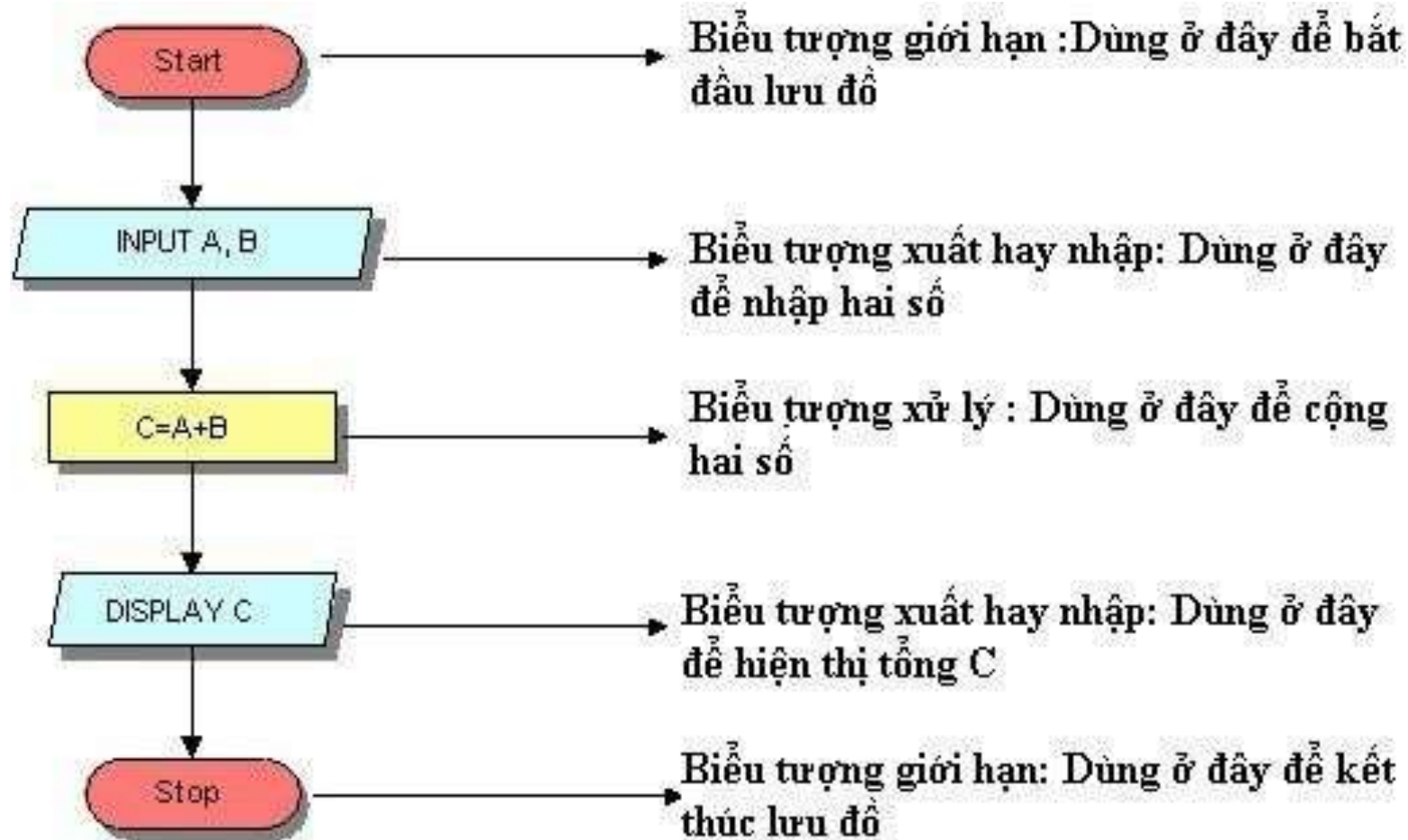
```
BEGIN
INPUT a, b
IF a = 0 THEN
    IF b = 0 THEN
        PRINT "Phương trình vô số nghiệm"
    ELSE
        PRINT "Phương trình vô nghiệm"
    END IF
ELSE
    PRINT "Phương trình có nghiệm  $x = -b/a$ "
END IF
END
```

Lưu đồ (Flowchart)

- Lưu đồ mô tả giải thuật bằng các sơ đồ hình khối, mỗi khối quy định một hành động.

Biểu Tượng	Mô Tả
	Bắt đầu hay kết thúc chương trình
	Những bước tính toán
	Các lệnh xuất hay nhập
	Quyết định và rẽ nhánh
	Bộ nối hai phần trong chương trình (đầu nối)
	Dòng chảy

Lưu đồ: Ví dụ tính tổng hai số



Demo

Cấu trúc lựa chọn

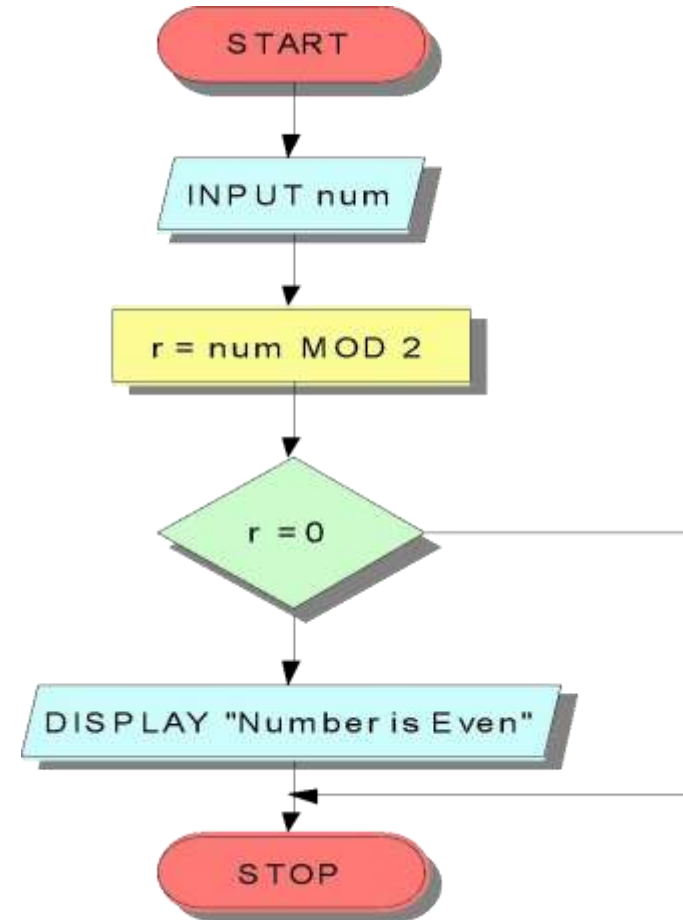
Cấu trúc lặp

Cấu trúc lựa chọn

- Lựa chọn một công việc để thực hiện căn cứ vào một điều kiện nào đó.
- Có một số dạng cơ bản như sau:
 - Cấu trúc 1: **Nếu** < điều kiện> (đúng) **thì** thực hiện < công việc>
 - Cấu trúc 2: **Nếu** < điều kiện> (đúng) **thì** thực hiện < công việc 1>, **ngược lại** (điều kiện sai) **thì** thực hiện < công việc 2>
 - Cấu trúc 3: Trường hợp < i> thì thực hiện < công việc i>

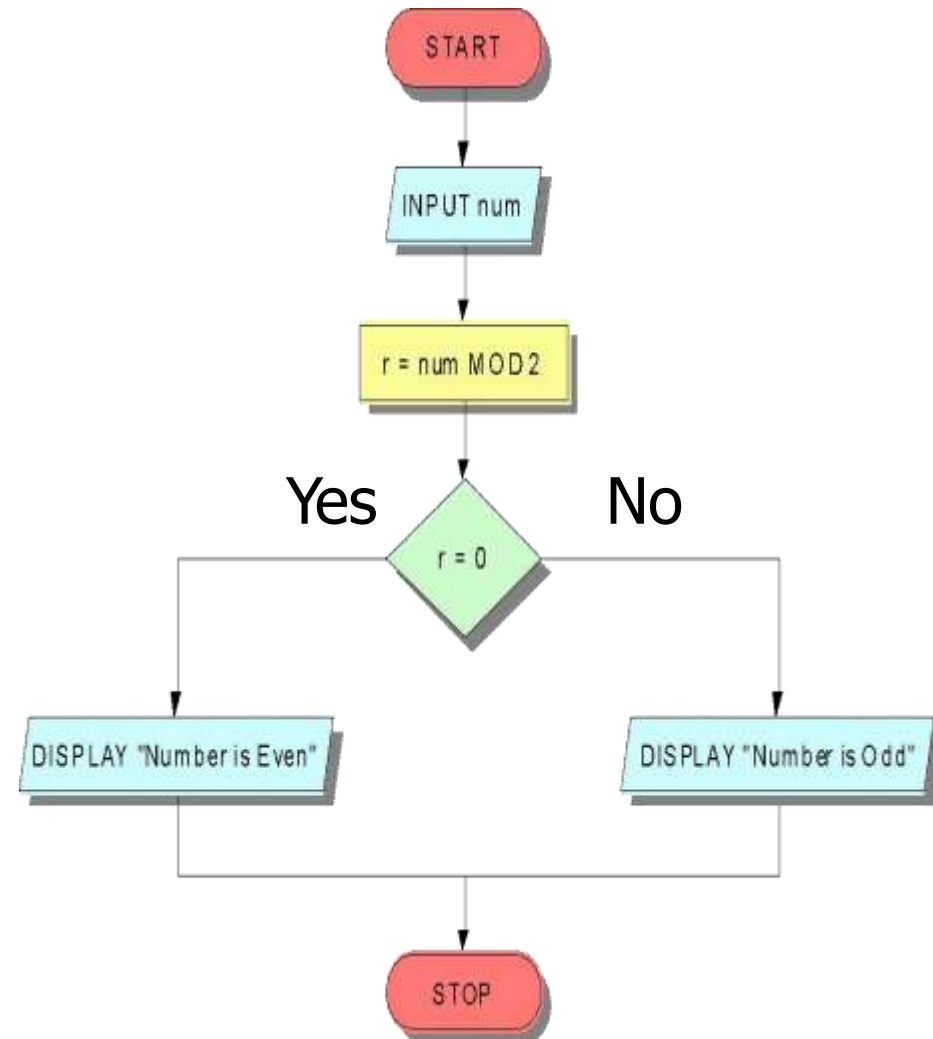
Ví dụ cấu trúc lựa chọn: Kiểm tra số chẵn

```
BEGIN  
INPUT num  
r = num MOD 2  
IF r=0  
    DISPLAY "Number is  
    even"  
END IF  
END
```



Cấu trúc lựa chọn: Kiểm tra số chẵn / lẻ

```
BEGIN  
INPUT num  
r=num MOD 2  
IF r=0  
    DISPLAY “Even Number”  
ELSE  
    DISPLAY “Odd Number”  
END IF  
END
```

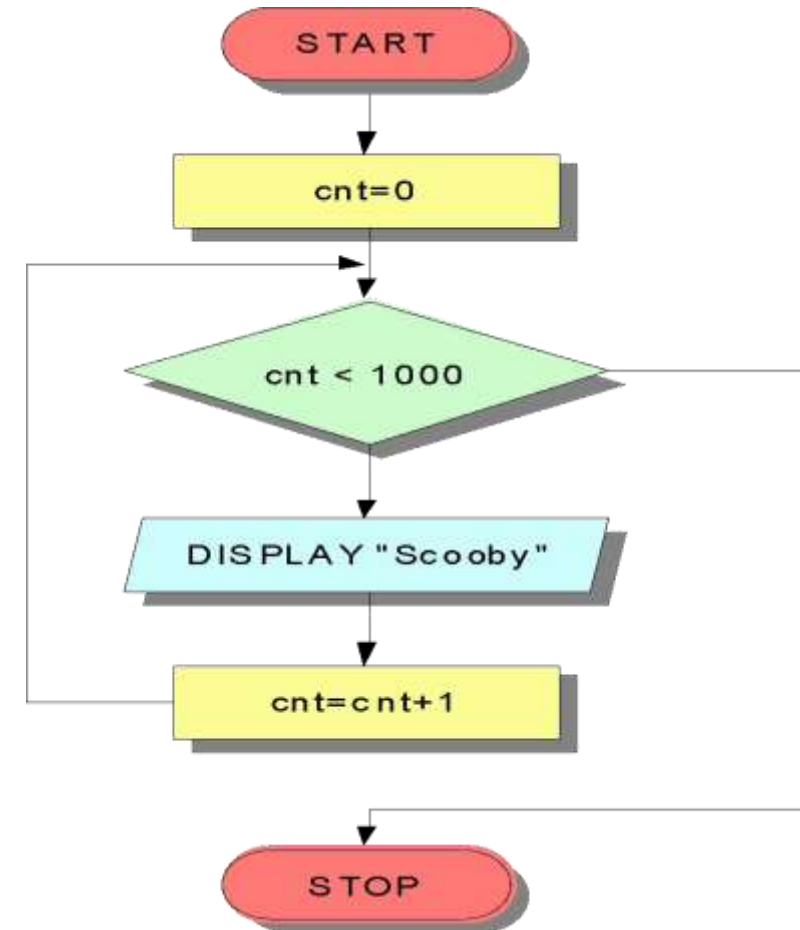


Cấu trúc lặp

- Thực hiện lặp lại một công việc không hoặc nhiều lần căn cứ vào một điều kiện nào đó.
- Có hai dạng như sau:
 - Lặp xác định: Biết trước số lần lặp
 - Lặp không xác định: Không biết trước số lần lặp

Cấu trúc lặp: Hiển thị 1000 từ Scooby

```
BEGIN  
cnt=0  
WHILE (cnt < 1000)  
DO  
    DISPLAY "Scooby"  
    cnt=cnt+1  
END DO  
END
```



Thảo luận

IDE - Integrated Development Environment

IDE – Integrated Development Environment

- Integrated Development Environment: Môi trường phát triển tích hợp là một loại phần mềm máy tính có công dụng giúp đỡ các lập trình viên trong việc phát triển phần mềm.
- IDE gồm:
 - Một trình soạn thảo (*source code editor*)
 - Trình biên dịch (*compiler*) và/hoặc trình thông dịch (*interpreter*).
 - Công cụ xây dựng tự động: khi sử dụng sẽ biên dịch (hoặc thông dịch) mã nguồn, thực hiện liên kết (*linking*), và có thể chạy chương trình một cách tự động.
 - Trình gỡ lỗi (*debugger*): hỗ trợ dò tìm lỗi.

IDE phổ biến cho Python



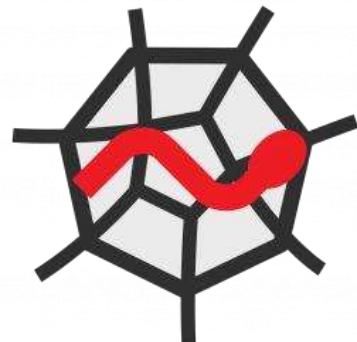
Visual Studio Code



Sublime Text



Pycharm



SPYDER

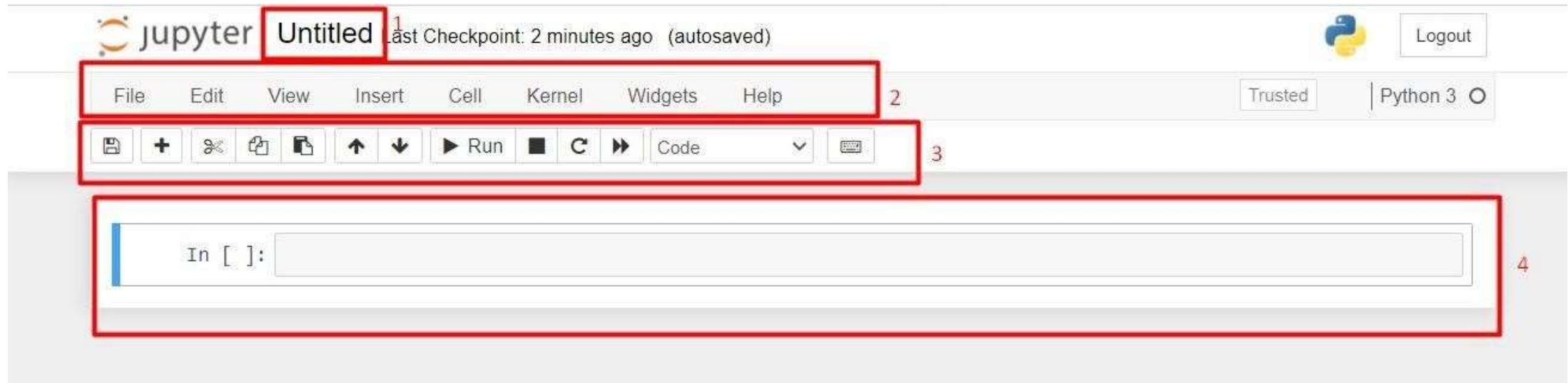


Atom/Atom-IDE



ANACONDA NAVIGATOR

Jupyter Notebook



Tổng kết

Ngôn ngữ lập trình

Thuật toán

Pseudo-code & Flowchart

Jupyter Notebook

Hướng dẫn

Hướng dẫn làm bài thực hành và bài tập

Chuẩn bị bài tiếp theo