

TRƯỜNG ĐÀO TẠO LẬP TRÌNH VIÊN VÀ QUẢN TRỊ MẠNG QUỐC TẾ BACHKHOA-APTECH

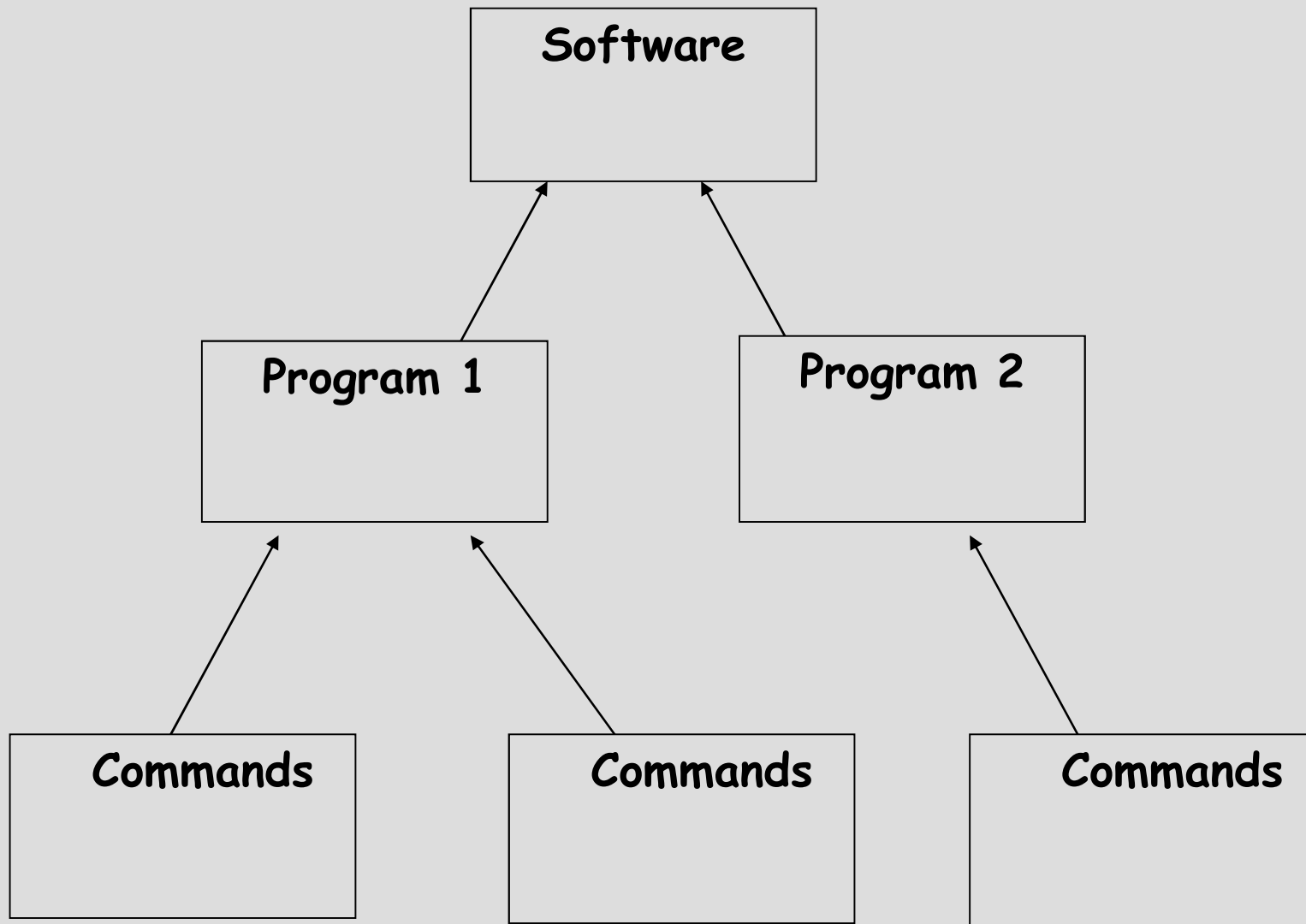
---

# BÀI 1: KHÁI NIỆM CƠ BẢN C

# MỤC TIÊU

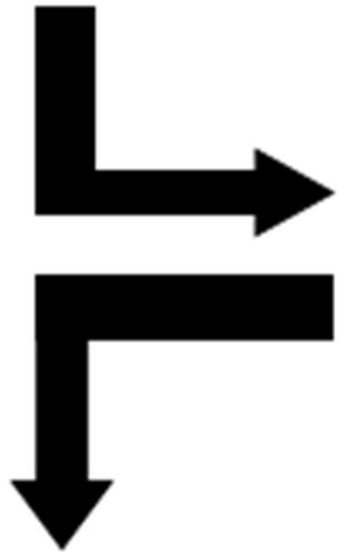
- Phân biệt sự khác nhau giữa câu lệnh, chương trình và phần mềm
- Quá trình hình thành ngôn ngữ C
- Khái niệm ngôn ngữ C
- Cấu trúc ngôn ngữ C
- Thư viện C
- Biên dịch và thi hành chương trình
- Các bước giải quyết vấn đề
- Khái niệm giải thuật (algorithms)
- Lưu đồ (flowchart) và các ký hiệu dùng trong lưu đồ

# CÂU LỆNH, CHƯƠNG TRÌNH, PHẦN MỀM



# QUÁ TRÌNH HÌNH THÀNH NGÔN NGỮ C

BPCL – Martin Richards



B – Ken Thompson

C – Dennis Ritchie



Bell Laboratories, Inc.

# CÁC LĨNH VỰC ỨNG DỤNG CỦA C

C được dùng để lập trình hệ thống

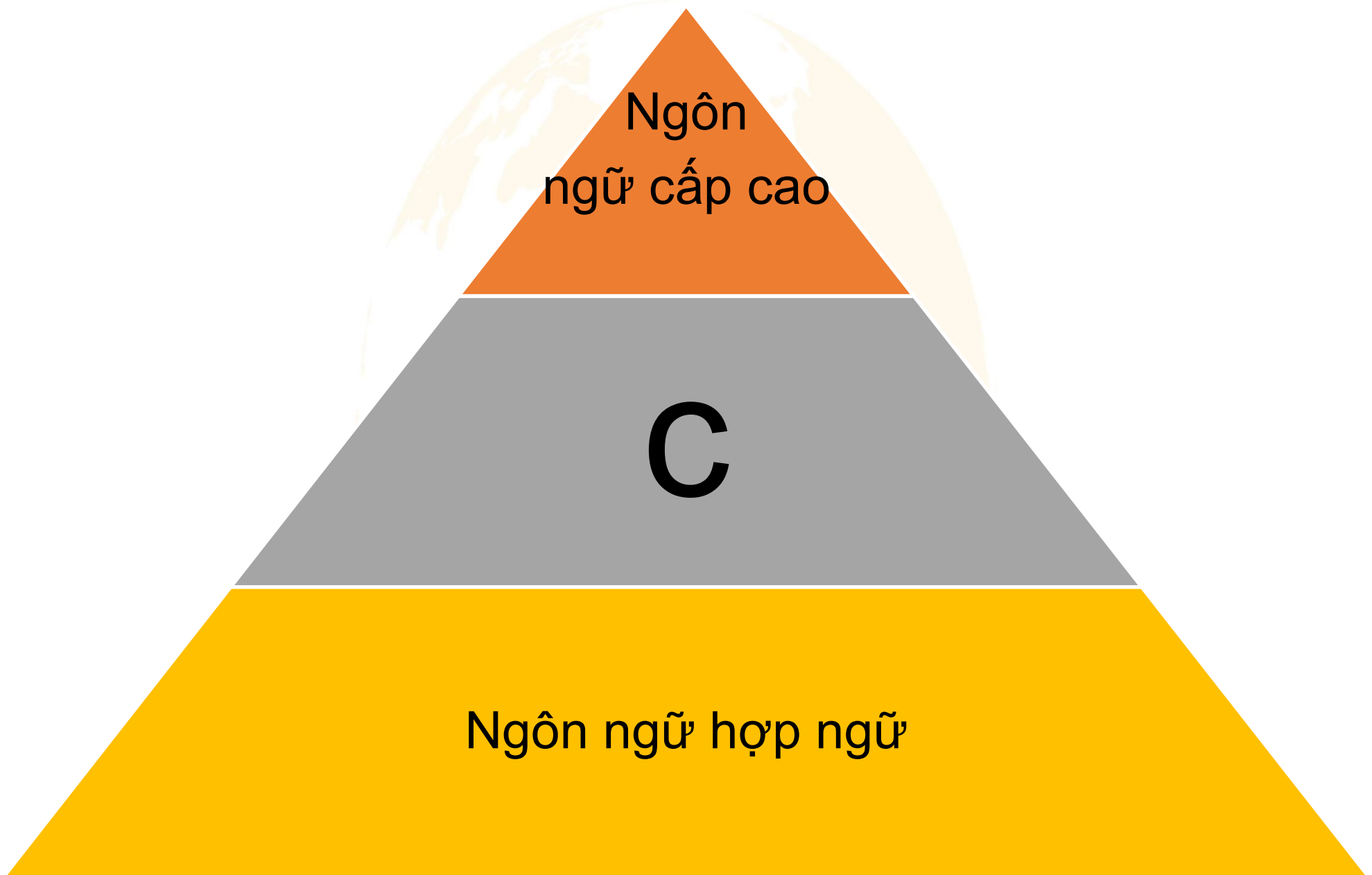
Một chương trình hệ thống làm thành một phần hệ điều hành hoặc các tiện ích hỗ trợ của hệ điều hành

Hệ điều hành (Operating Systems), trình thông dịch (Interpreters), trình soạn thảo (Editors), trình Hợp Ngữ (Assembly) được gọi là chương trình hệ thống

Hệ điều hành UNIX được phát triển dựa vào C

Có các trình biên dịch dành cho hầu hết các loại hệ thống PC

# KHÁI NIỆM C – NGÔN NGỮ CẤP TRUNG



# KHÁI NIỆM C – NGÔN NGỮ CÓ CẤU TRÚC

C cho phép tổng hợp mã lệnh và dữ liệu

Nó có khả năng tập hợp và ẩn đi tất cả thông tin, lệnh khởi phần còn lại của chương trình để dùng cho những tác vụ riêng

Chương trình C có thể được chia nhỏ thành những hàm (functions) hay những khối mã (code blocks)

```
do
{
    i = i + 1;
    .
    .
    .
} while (i < 40);
```

# KHÁI NIỆM C – ĐẶC ĐIỂM CỦA C

- ✓ **C** có 32 từ khóa
- ✓ Những từ khóa này kết hợp với cú pháp của C hình thành ngôn ngữ C
- ✓ Các quy tắc được áp dụng cho các chương trình C
  - ✓ Tất cả từ khóa là chữ thường
  - ✓ Đoạn mã trong chương trình C có phân biệt chữ thường, chữ hoa, do while khác DO WHILE
  - ✓ Từ khóa không thể dùng đặt tên biến (variable name) hoặc tên hàm (function name)

```
main()  
{  
/* This is a sample Program*/  
    int i,j;  
    i=100;  
    j=200;  
    :  
}
```



# CẤU TRÚC CHƯƠNG TRÌNH C 1-4

## main()

- Chương trình C được chia nhỏ thành những đơn vị gọi là hàm
- Không kể có bao nhiêu hàm trong chương trình, hệ điều hành luôn trao quyền điều khiển cho hàm main() khi một chương trình C được thực thi.
- Theo sau tên hàm là dấu ngoặc đơn
- Dấu ngoặc đơn có thể có chứa hay không chứa những tham số

## Dấu phân cách {...}

- Sau phần đầu hàm là dấu ngoặc xoắn mở {
- Nó cho biết việc thi hành lệnh trong hàm bắt đầu
- Tương tự, dấu ngoặc xoắn đóng } sau câu lệnh cuối cùng trong hàm chỉ ra điểm kết thúc của hàm

## Dấu kết thúc câu lệnh ...;

- Một câu lệnh trong C được kết thúc bằng dấu chấm phẩy ;
- Trình biên dịch C không hiểu việc xuống dòng, khoảng trắng hay tab
- Một câu lệnh không kết thúc bằng dấu chấm phẩy sẽ được xem như dòng lệnh lỗi trong C

# CẤU TRÚC CHƯƠNG TRÌNH C 4-4

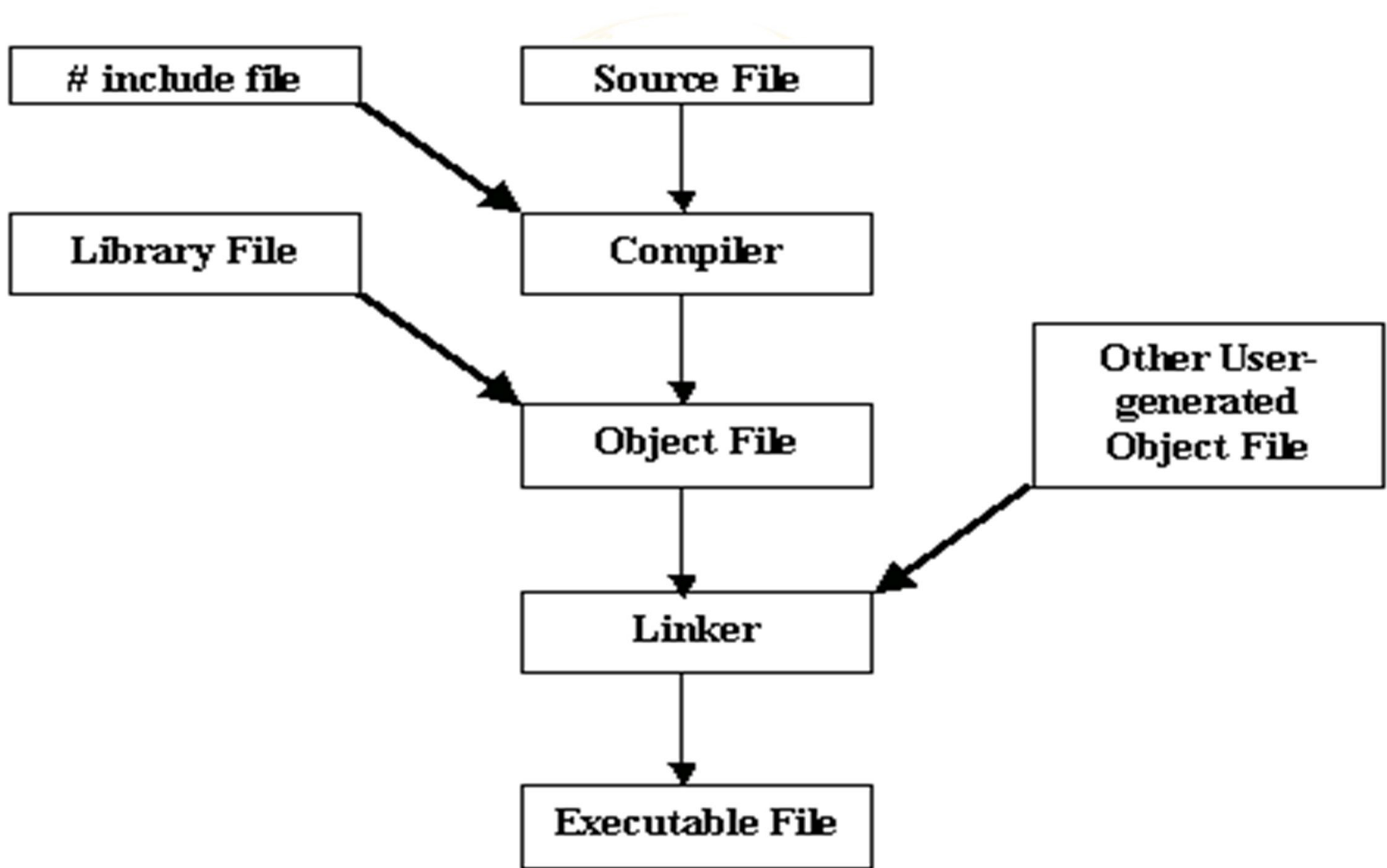
## /\*Dòng chú thích\*/

- Những chú thích thường được viết để mô tả công việc của một lệnh đặc biệt, một hàm hay toàn bộ chương trình
- Trình biên dịch sẽ bỏ qua phần chú thích
- Trong trường hợp chú thích nhiều dòng, nó sẽ bắt đầu bằng ký hiệu /\* và kết thúc là \*/



- ✓ Tất cả trình biên dịch C đều chứa một thư viện hàm chuẩn
- ✓ Một hàm được viết bởi lập trình viên có thể được đặt trong thư viện và được dùng khi cần thiết
- ✓ Một số trình biên dịch cho phép thêm hàm vào thư viện chuẩn
- ✓ Một số trình biên dịch yêu cầu tạo một thư viện riêng

# BIÊN DỊCH VÀ THI HÀNH CHƯƠNG TRÌNH



# CÁC BƯỚC GIẢI QUYẾT VẤN ĐỀ 1-2

Giải thuật gồm một tập hợp các bước thực hiện nhằm giải quyết một vấn đề. Thí dụ sau đây mô tả một giải thuật

Xuống  
tầng hầm

Đi đến quán  
ăn tự phục vụ

Đến cầu thang

Rời phòng học

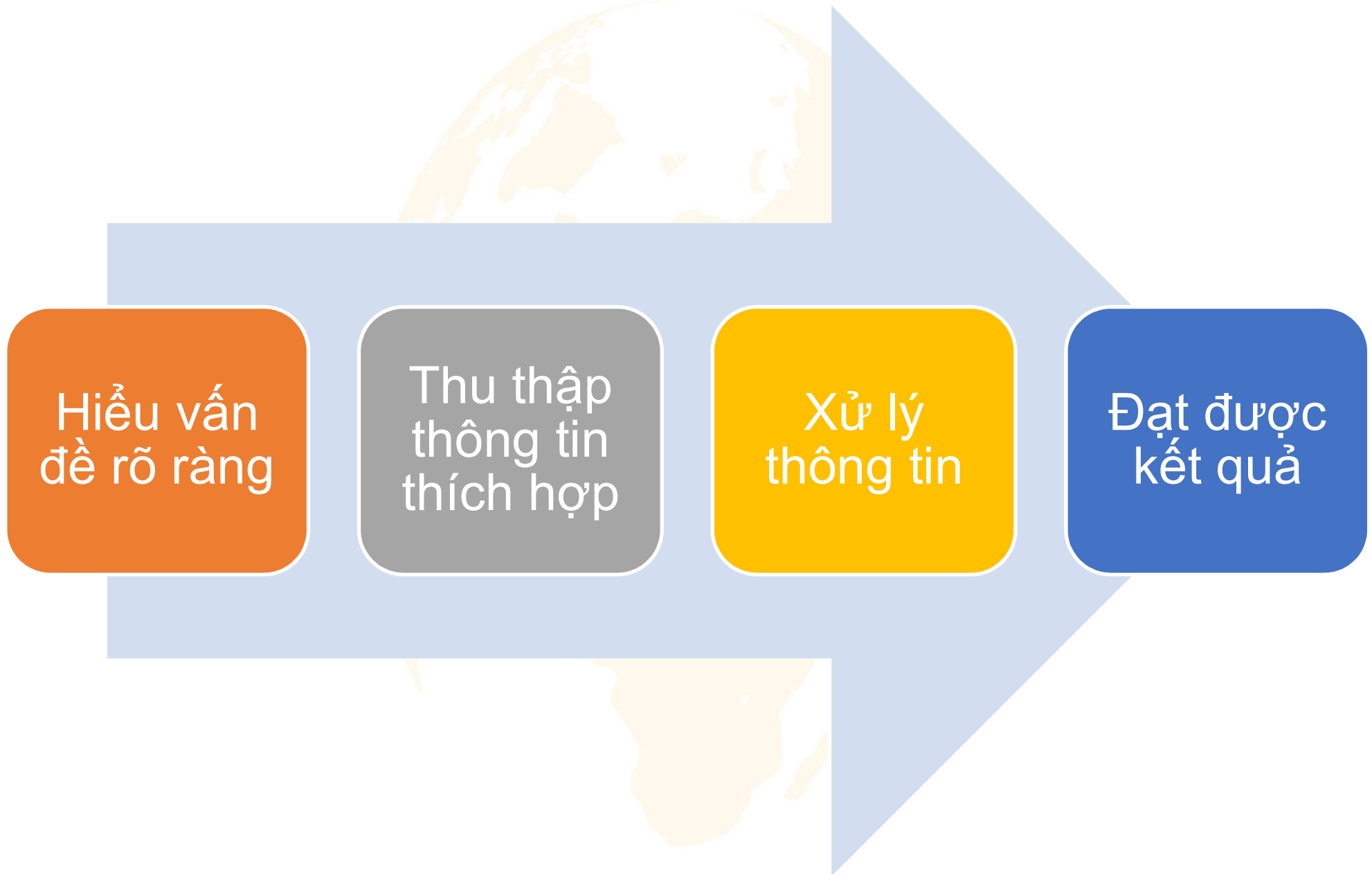
Phòng học



Cafeteria

**Đây là các bước thực hiện khi một người muốn đi đến quán ăn tự phục vụ từ phòng học**

# CÁC BƯỚC GIẢI QUYẾT VẤN ĐỀ 2-2





# MÃ GIẢ (PSEUDOCODE)

- ✓ Không là mã thật. Một phương pháp viết giải thuật sử dụng một tập hợp các từ tương tự mã thật
- ✓ Mỗi đoạn mã giả phải bắt đầu với một từ BEGIN
- ✓ Để hiển thị giá trị nào đó, từ DISPLAY được dùng
- ✓ Mã giả kết thúc với từ END

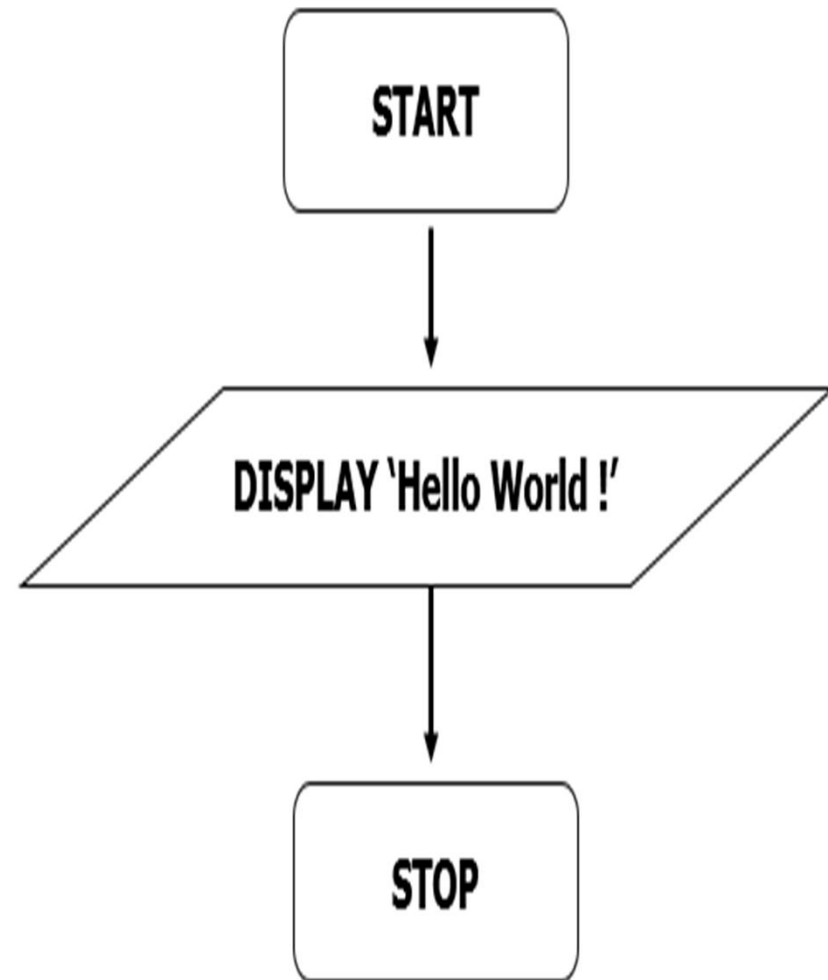
**BEGIN**

**DISPLAY 'Hello World !'**


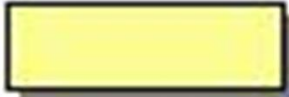




**END**

# LƯU ĐỒ (FLOWCHARTS)

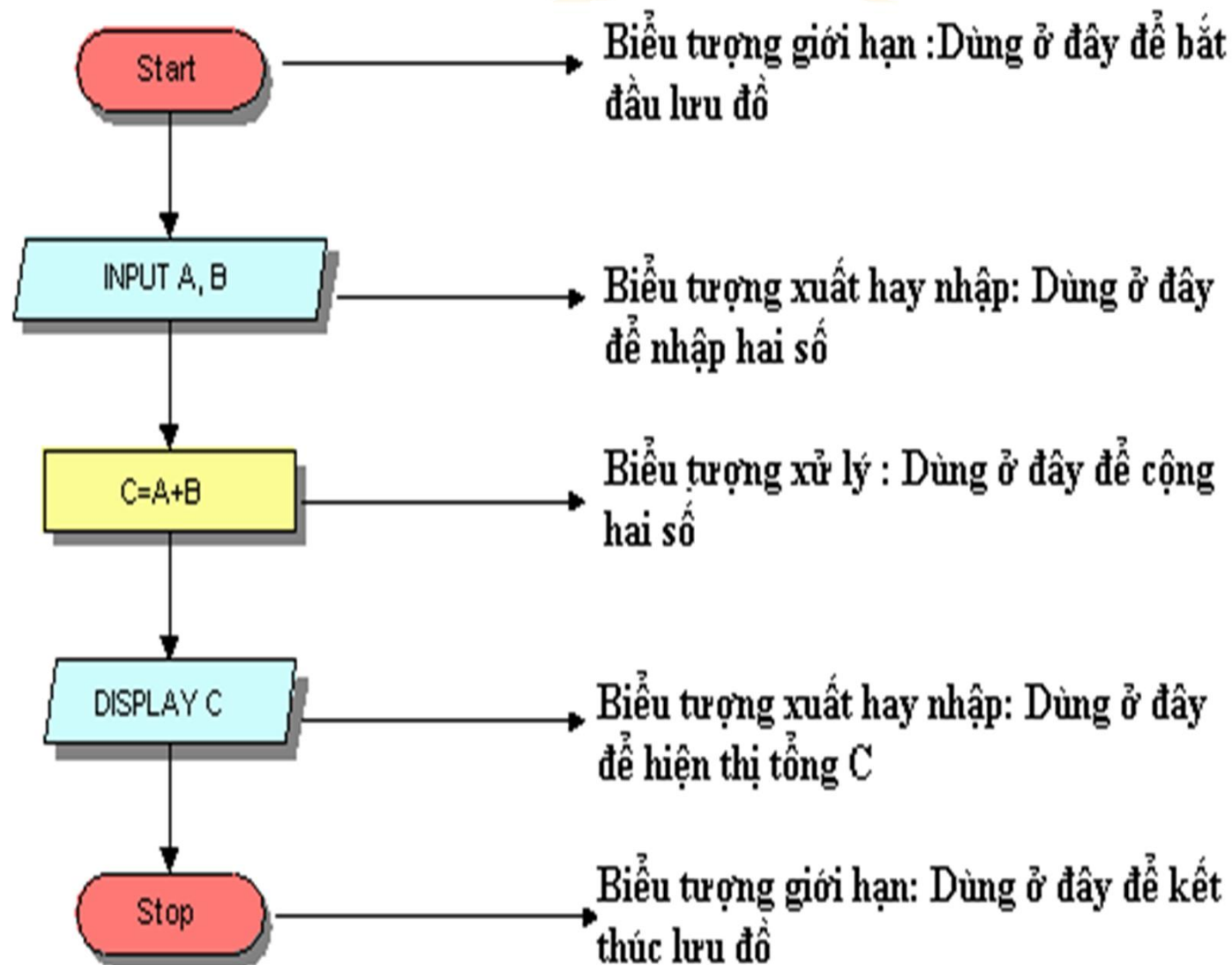
Lưu đồ là một hình ảnh minh  
hoạ cho giải thuật



# BIỂU TƯỢNG TRONG LƯỚI ĐỒ

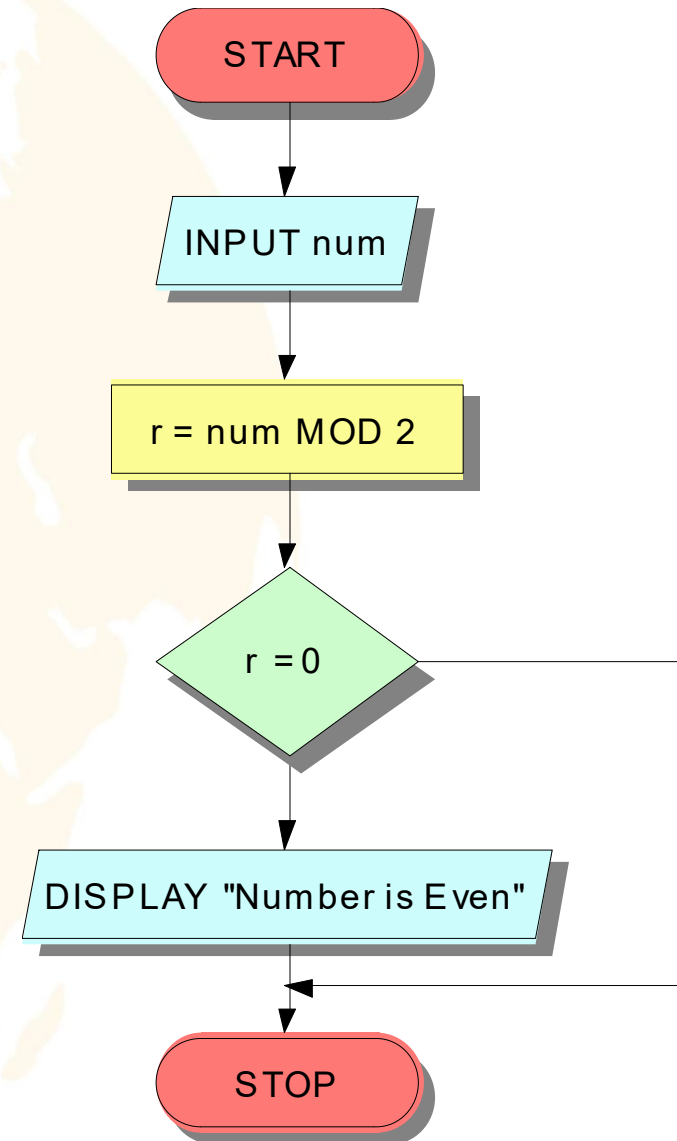
Biểu Tượng	Mô Tả
	Bắt đầu hay kết thúc chương trình
	Những bước tính toán
	Các lệnh xuất hay nhập
	Quyết định và rẽ nhánh
	Bộ nối hai phần trong chương trình (đầu nối)
	Dòng chảy

# LƯU ĐỒ CỘNG HAI SỐ



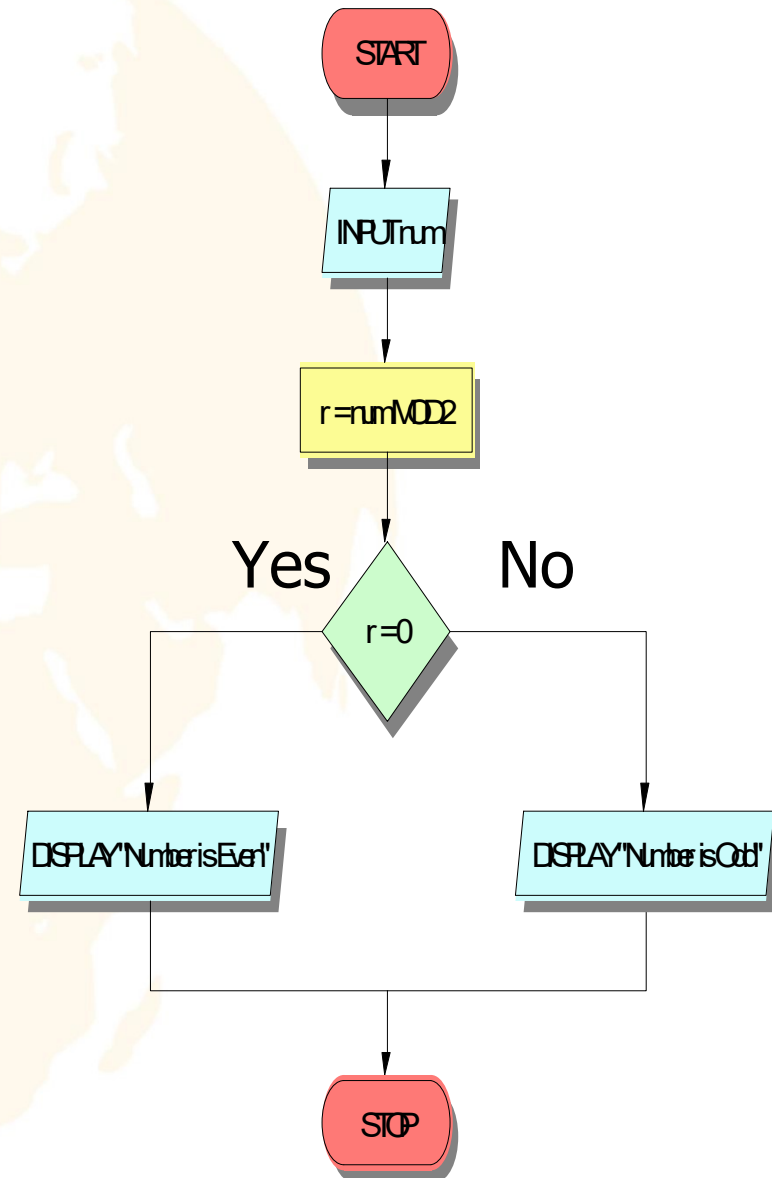
# CẤU TRÚC IF

```
BEGIN
INPUT num
r = num MOD 2
IF r=0
    Display “Number is
even”
END IF
END
```



# CẤU TRÚC IF...ELSE

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



# ĐA ĐIỀU KIỆN SỬ DỤNG AND/OR

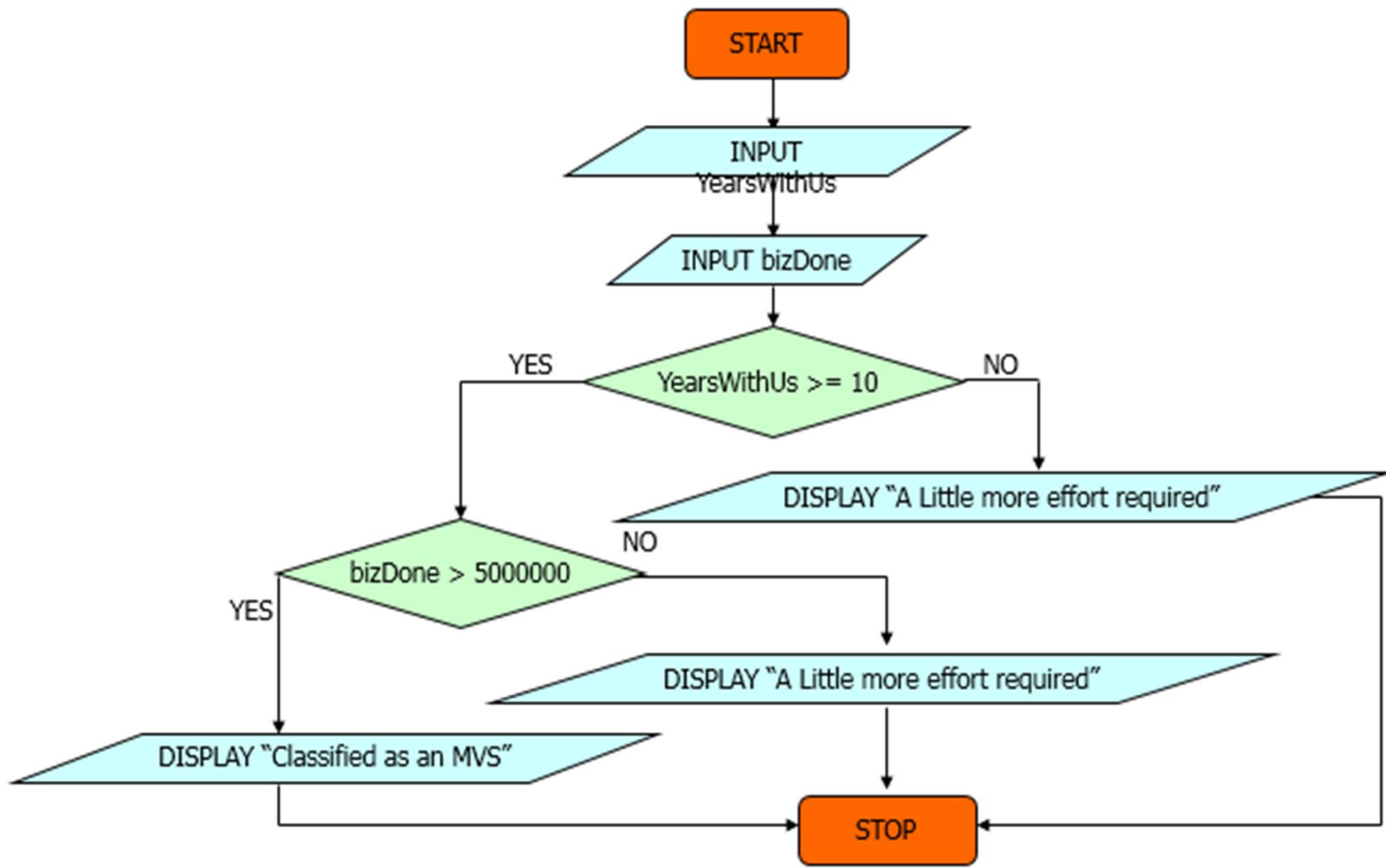
```
BEGIN
INPUT yearsWithUs
INPUT bizDone
IF yearsWithUs >= 10 AND bizDone >=5000000
    DISPLAY “Classified as an MVS”
ELSE
    DISPLAY “A little more effort required!”
END IF
END
```

# CẤU TRÚC IF LỒNG NHAU 1-2

```
BEGIN
INPUT yearsWithUs
INPUT bizDone
IF yearsWithUs >= 10
    IF bizDone >= 5000000
        DISPLAY "Classified as an MVS"
    ELSE
        DISPLAY "A little more effort required!"
END IF
ELSE
    DISPLAY "A little more effort required!"
END IF
END
```

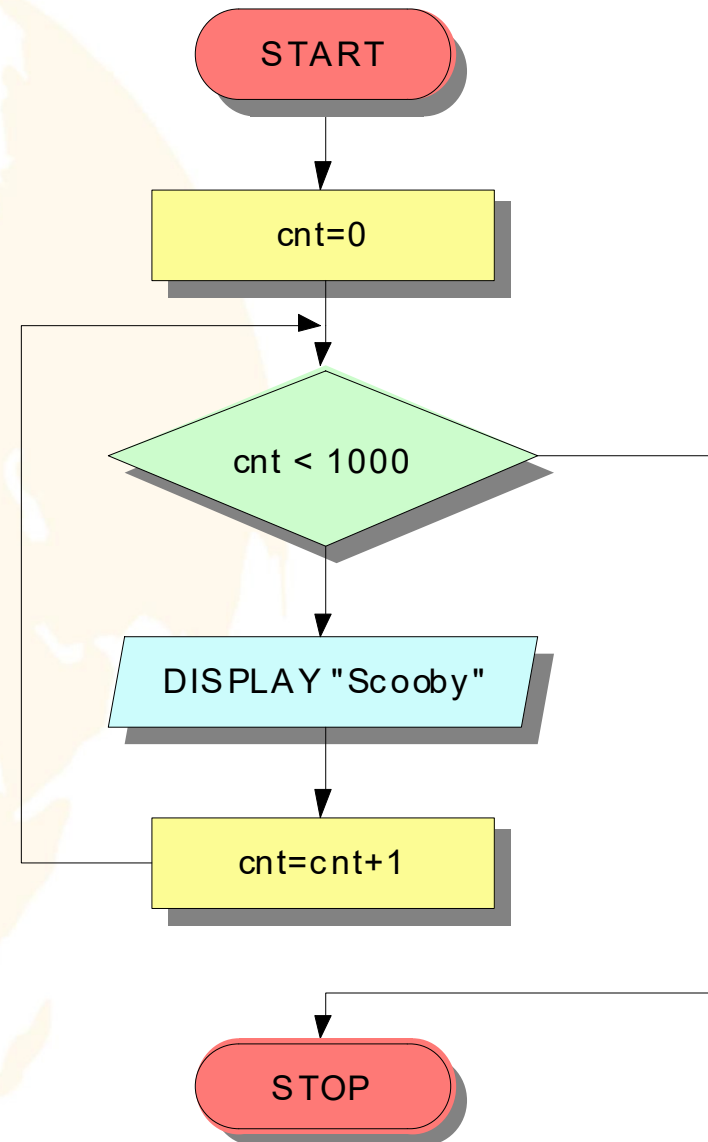


# CẤU TRÚC IF LỒNG NHAU 2-2



# VÒNG LẶP

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



# TÓM TẮT BÀI HỌC

- ✓ Phần mềm là một tập hợp các chương trình, một chương trình là một tập hợp các chỉ thị (lệnh)
- ✓ Ngôn ngữ C có 32 từ khóa
- ✓ Các bước cần thiết để giải quyết một bài toán
  - Nghiên cứu chi tiết bài toán
  - Thu thập thông tin thích hợp
  - Xử lý thông tin
  - Kết quả
- ✓ Một giải thuật là một danh sách rút gọn và logic các bước để giải quyết vấn đề
- ✓ Cấu trúc lệnh chọn lựa, điều kiện và cấu trúc rẽ nhánh
- ✓ Cấu trúc vòng lặp

TRƯỜNG ĐÀO TẠO LẬP TRÌNH VIÊN VÀ QUẢN TRỊ MẠNG QUỐC TẾ BACHKHOA-APTECH

# THANK FOR WATCH!

