



Cơ bản của C

Phiên 1

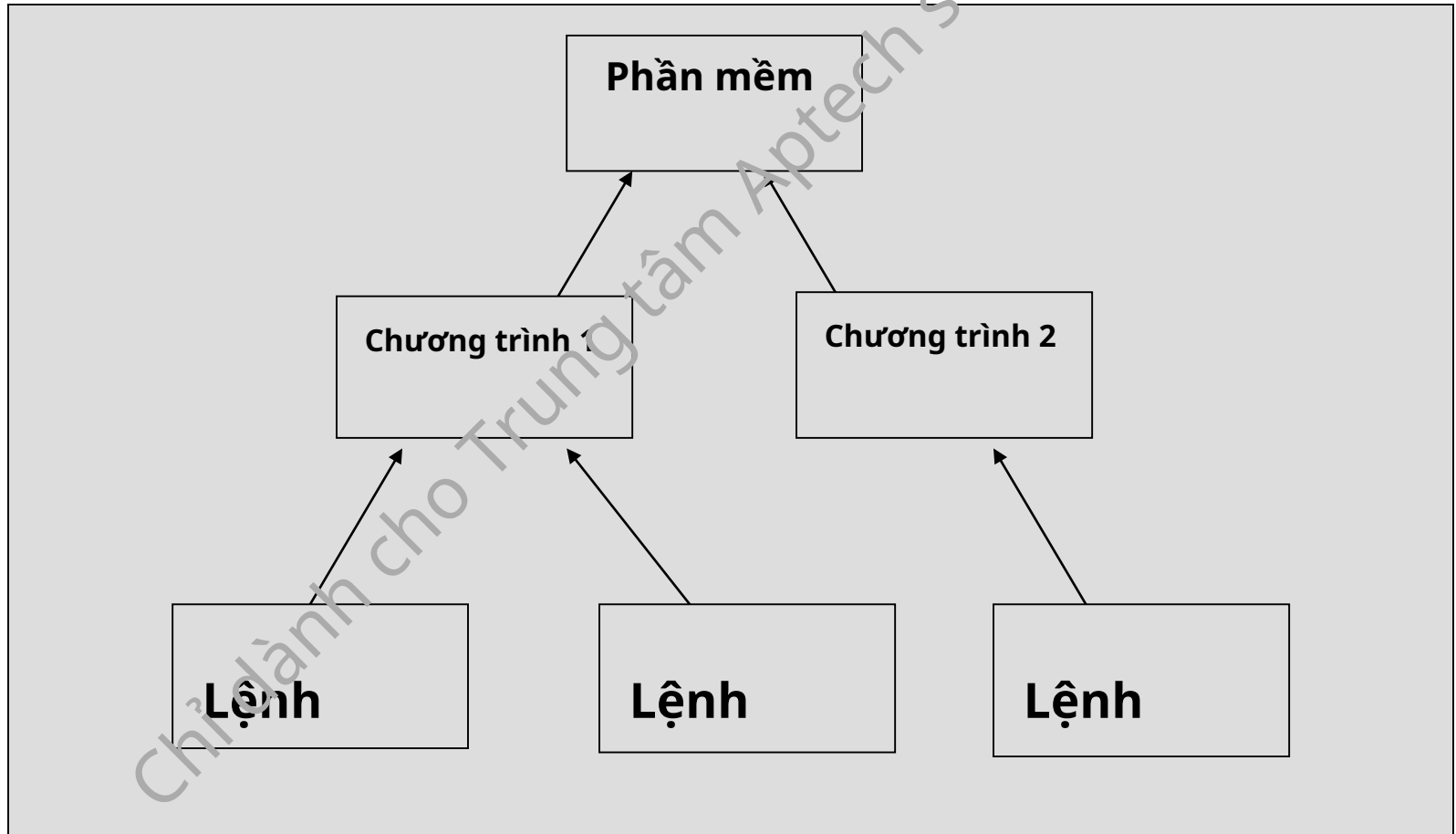
Chỉ dành cho Trung tâm Aptech sử dụng



Mục tiêu

- Phân biệt giữa Command, Program và Software
- Giải thích sự khởi đầu của C
- Giải thích khi nào và tại sao C được sử dụng
- Thảo luận về cấu trúc chương trình C
- Thảo luận về thuật toán
- Vẽ sơ đồ luồng công việc
- Liệt kê các ký hiệu được sử dụng trong sơ đồ luồng

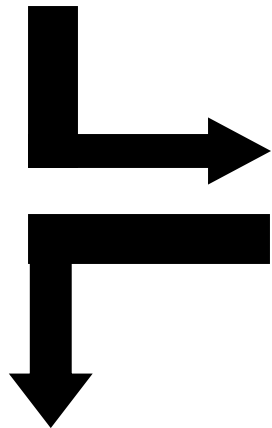
Phần mềm, Chương trình và Yêu cầu





Sự khởi đầu của C

BPCL – Martin Richards



B – Ken Thompson

C – Dennis Ritchie



Bell Laboratories, Inc.



Các lĩnh vực ứng dụng của C

- C ban đầu được sử dụng để lập trình hệ thống
- Một chương trình hệ thống tạo thành một phần của hệ điều hành máy tính hoặc các tiện ích hỗ trợ của nó
- Hệ điều hành, Trình thông dịch, Trình biên tập, Chương trình lắp ráp thường được gọi là chương trình hệ thống
- Hệ điều hành UNIX được phát triển bằng C
- Có trình biên dịch C có sẵn cho hầu hết các loại máy tính



Ngôn ngữ trình độ trung cấp

Ngôn ngữ cấp cao



Ngôn ngữ lắp ráp



Ngôn ngữ có cấu trúc

- C cho phép phân chia mã và dữ liệu

- Nó đề cập đến khả năng cắt bỏ và ẩn tất cả thông tin và hướng dẫn cần thiết để thực hiện một nhiệm vụ cụ thể, từ phần còn lại của chương trình

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

- Mã có thể được phân chia thành C bằng cách sử dụng các hàm hoặc khối mã.



Về C

- **C** có 32 từ khóa
- Những từ khóa này kết hợp với cú pháp chính thức tạo thành một ngôn ngữ lập trình
- Các quy tắc cần tuân theo đối với tất cả các chương trình được viết bằng C:
- Tất cả các từ khóa đều được viết thường
- C phân biệt chữ hoa chữ thường, làm trong khi khác với LÀM TRONG KHI
- Từ khóa không thể được sử dụng làm tên biến hoặc tên hàm

```
chủ yếu()
{
/* Đây là một chương trình mẫu*/
    số nguyên i,j;
    tôi = 100;
    j=200;
    :
}
```




Chương trình C

Cấu trúc-1

main()

C programs are divided into units called functions.

Irrespective of the number of functions in a program, the operating system always passes control to `main()` when a C program is executed.

The function name is always followed by parentheses.

The parentheses may or may not contain parameters.



Chương trình C

Cấu trúc-2

Delimiters { ... }

The function definition is followed by an open curly brace (`{`).

This curly brace signals the beginning of the function.

Similarly a closing curly brace (`}`) after the codes, in the function, indicate the end of the function.



Chương trình C

Cấu trúc-3

Statement Terminator ;

A statement in C is terminated with a semicolon

A carriage return, whitespace, or a tab is not understood by the C compiler.

A statement that does not end in a semicolon is treated as an erroneous line of code in C.



Chương trình C

Cấu trúc-4

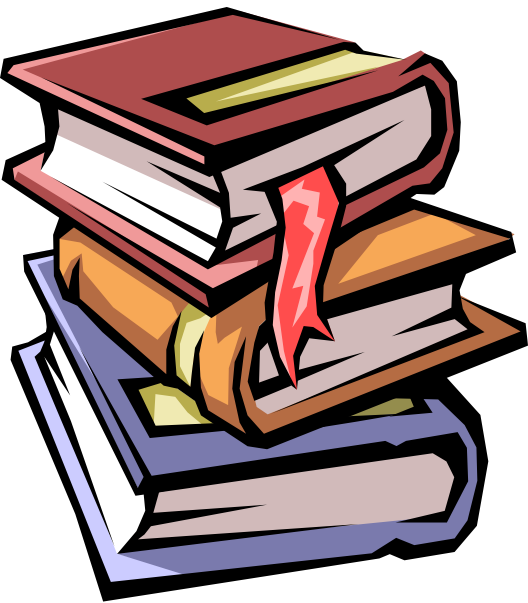
`/* Comment Lines */`

Comments are usually written to describe the task of a particular command, function or an entire program.

The compiler ignores them. In C, comments begin with `/*` and are terminated with `*/`, in case the comments contain multiple lines



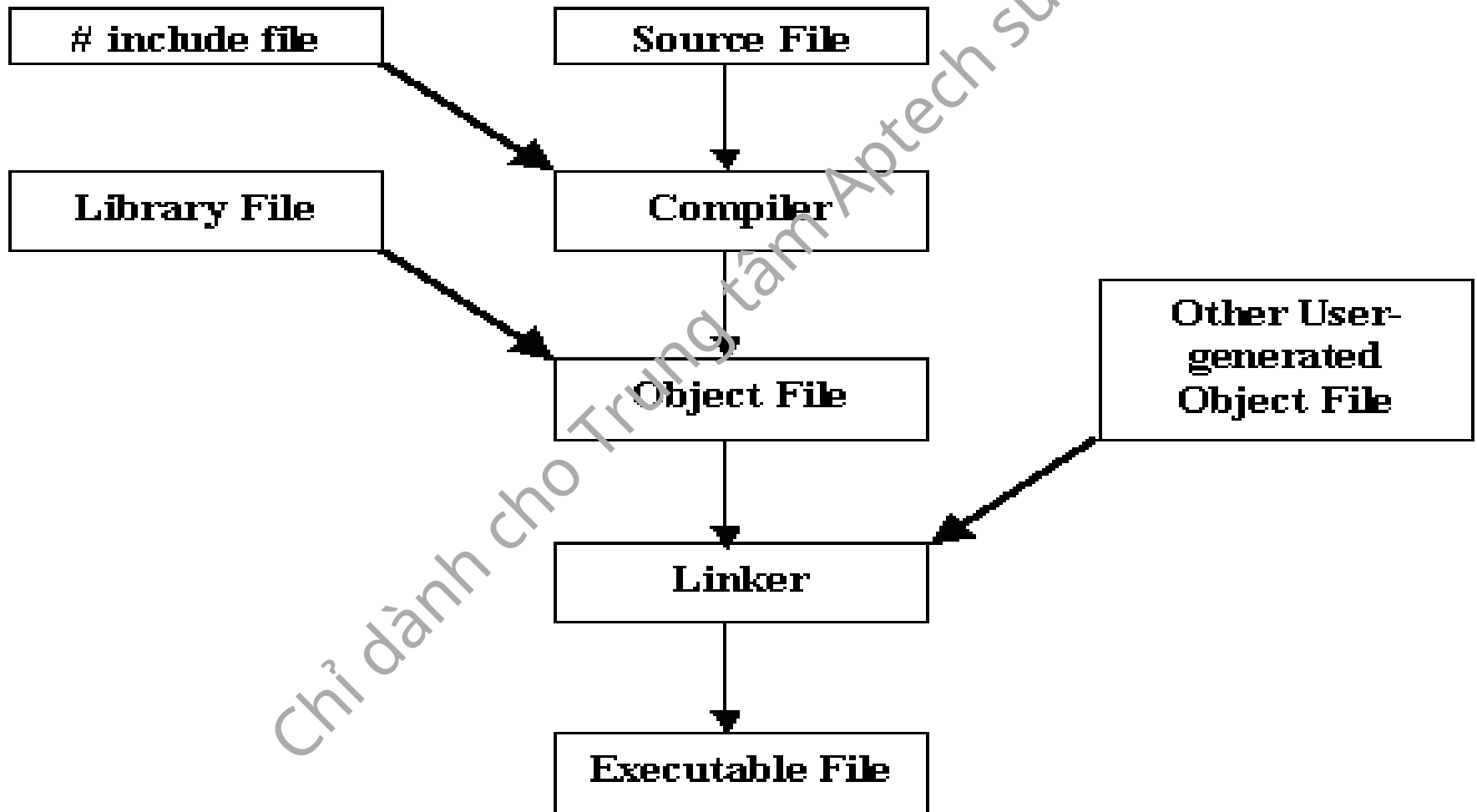
Thư viện C



- Tất cả các trình biên dịch C đều có một thư viện chuẩn các hàm
- Một hàm do lập trình viên viết có thể được đưa vào thư viện và sử dụng khi cần thiết
- Một số trình biên dịch cho phép thêm các hàm vào thư viện chuẩn
- Một số trình biên dịch yêu cầu phải tạo một thư viện riêng

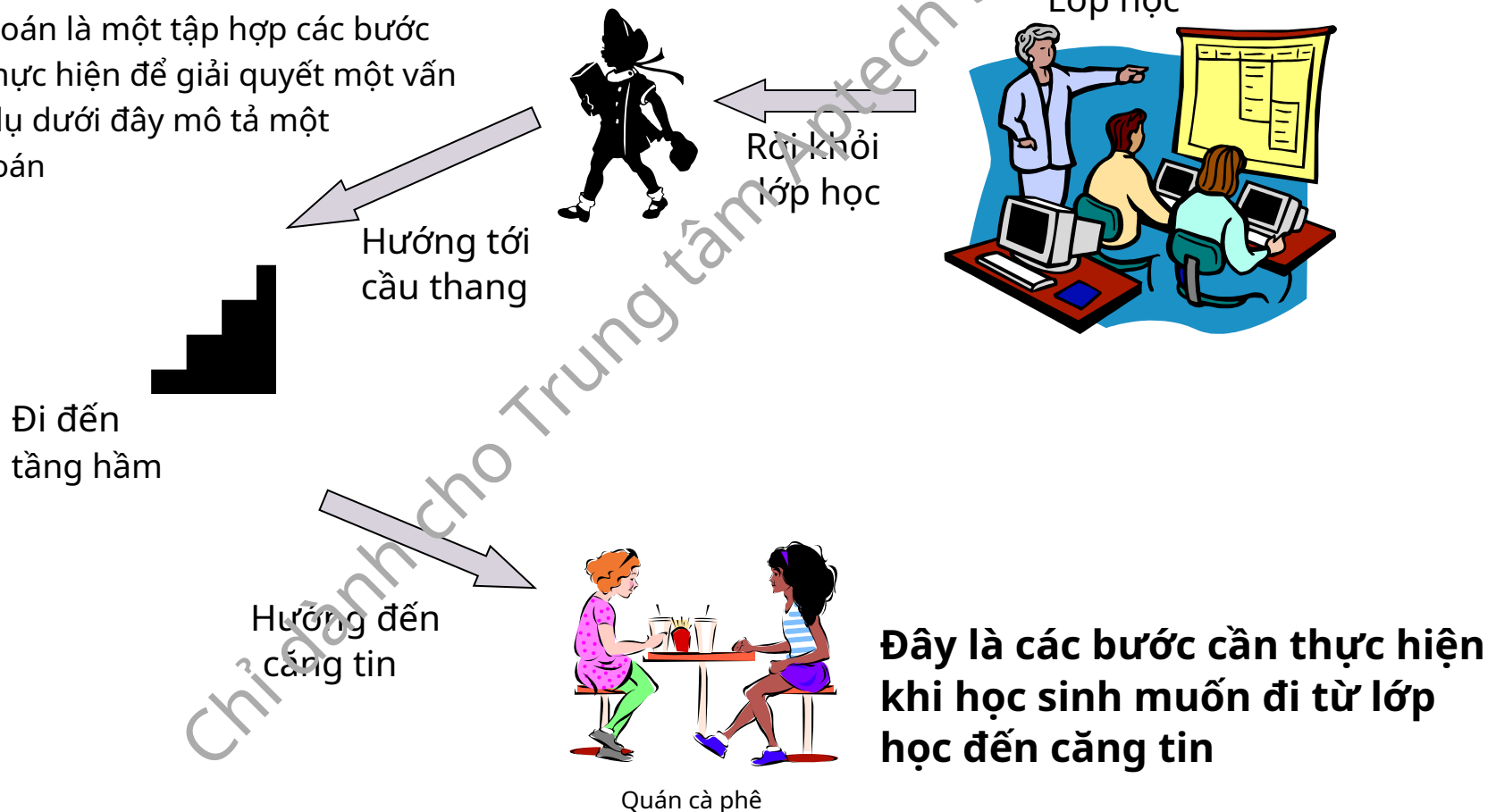
Biên dịch & Chạy A

Chương trình



Phương pháp lập trình để Giải quyết vấn đề

Thuật toán là một tập hợp các bước được thực hiện để giải quyết một vấn đề. Ví dụ dưới đây mô tả một thuật toán





Giải quyết một vấn đề

Để giải quyết một vấn đề

Hiểu rõ vấn đề

Thu thập thông tin có liên quan

Xử lý thông tin

Đến với giải pháp



Mã giả

Nó không phải là mã thực tế. Một phương pháp viết thuật toán sử dụng một tập hợp các từ chuẩn khiến nó giống với mã

BẮT ĐẦU

HIỂN THỊ 'Xin chào thế giới!'

KẾT THÚC

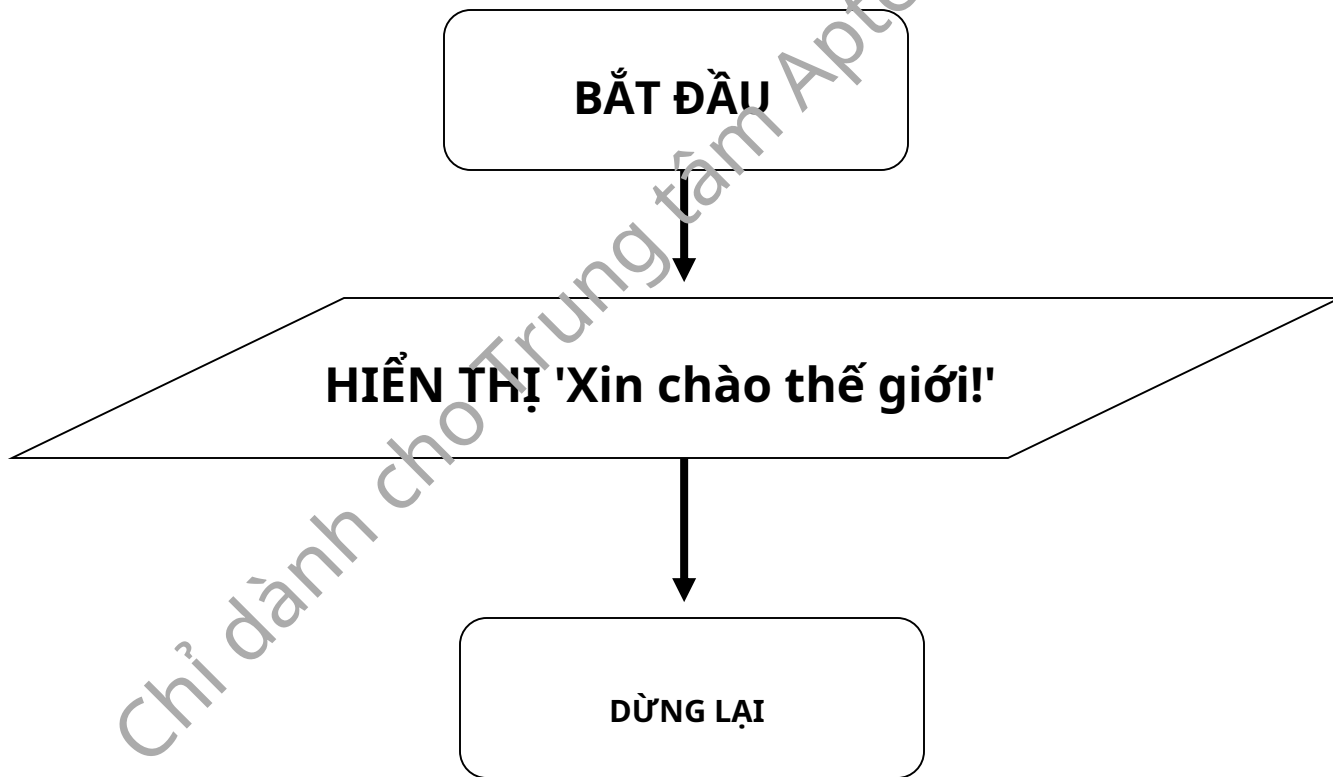
Mỗi mã giả bắt đầu bằng BEGIN

Để hiển thị một số giá trị, từ DISPLAY được sử dụng. Mã giả kết thúc bằng KẾT THÚC



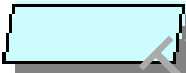

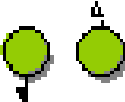



Biểu đồ luồng

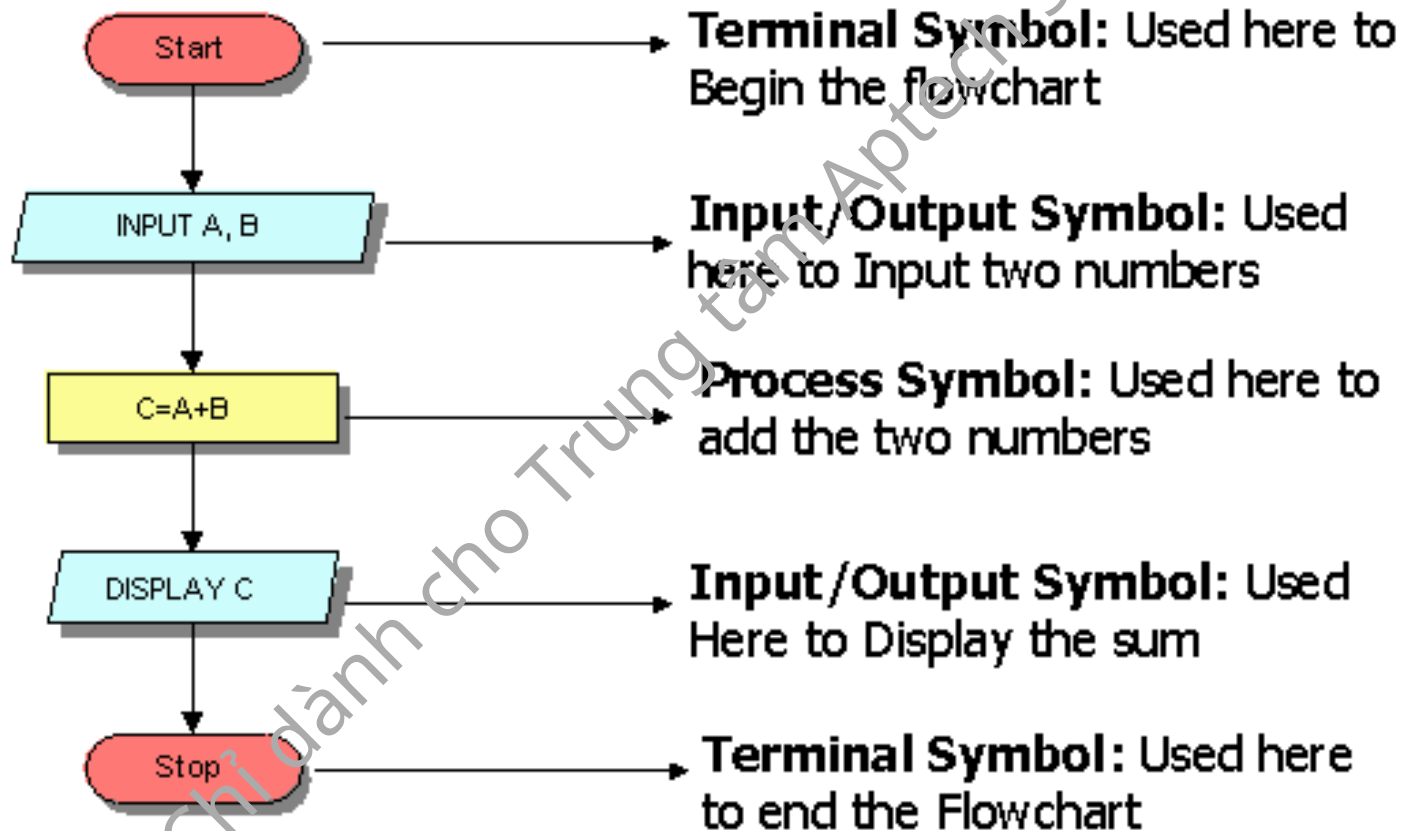
Nó là một biểu diễn đồ họa của một thuật toán



Biểu tượng sơ đồ dòng chảy

Symbol	Description
	Start or End of the Program
	Computational Steps
	Input / Output instructions
	Decision making & Branching
	Connectors
	Flow Line

Biểu đồ để thêm hai số



Cấu trúc IF

BẮT ĐẦU

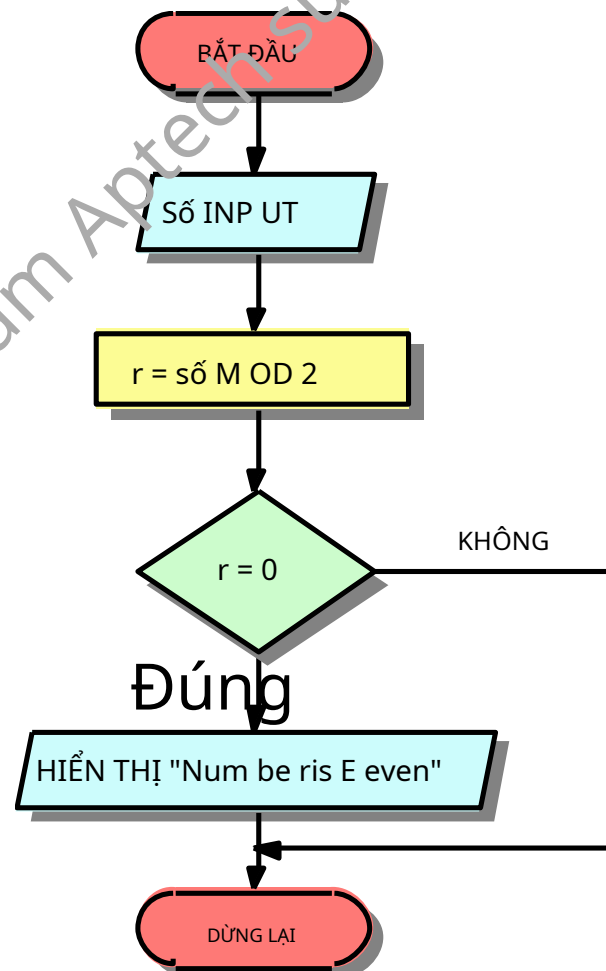
Số đầu vào

$r = \text{số MOD } 2$

NẾU NHƯ $r=0$

Hiển thị "Số chẵn" **KẾT THÚC NẾU**

KẾT THÚC



Cấu trúc IF-ELSE

BẮT ĐẦU

Số đầu vào

$r = \text{số} \text{ MOD } 2$

NẾU NHƯ $r=0$

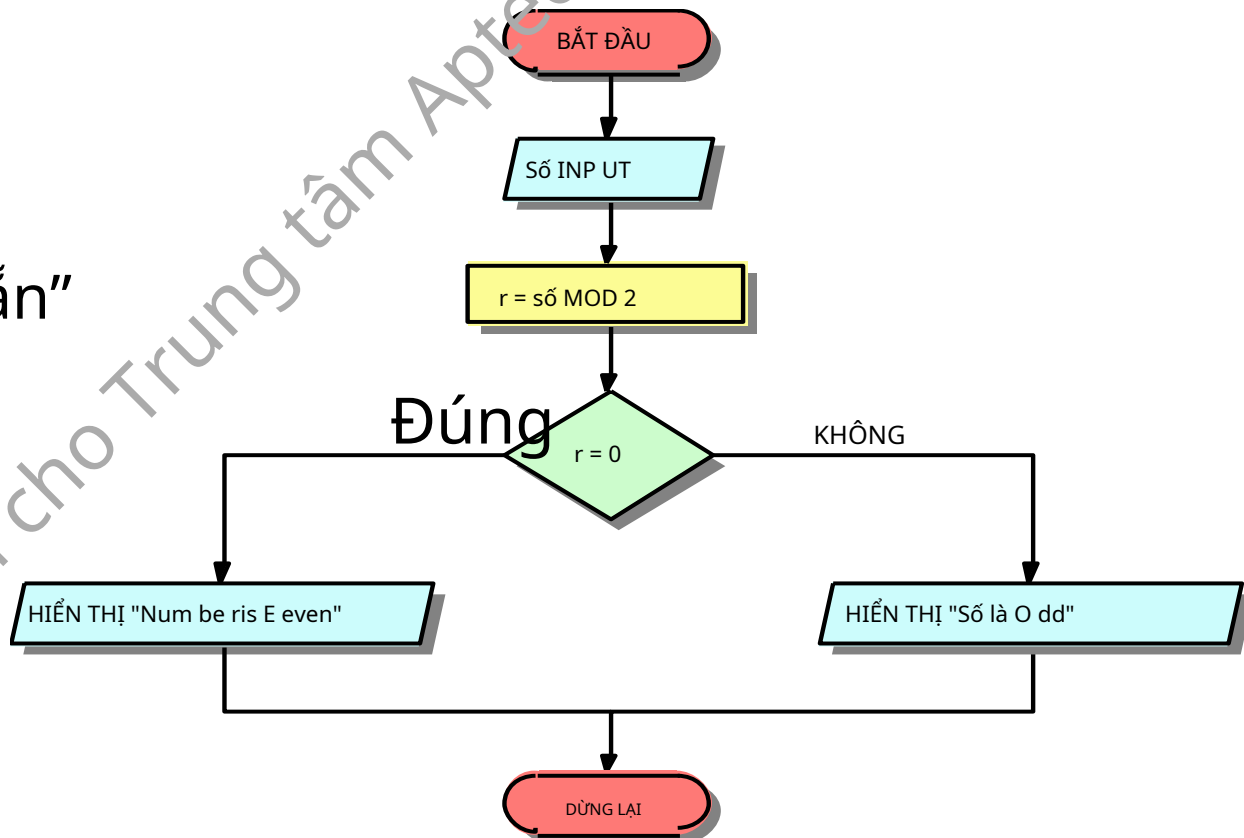
HIỂN THỊ "Số Chẵn"

KHÁC

HIỂN THỊ "Số Lẻ"

KẾT THÚC NẾU

KẾT THÚC





Nhiều tiêu chí sử dụng VÀ/HOẶC

BẮT ĐẦU

NHẬP NĂMVớichúngTôi

NHẬP bizDone

NẾU yearsWithUs ≥ 10 VÀ bizĐã hoàn thành ≥ 5000000

HIỂN THỊ “Được phân loại là MVS”

KHÁC

HIỂN THỊ “Cần thêm chút nỗ lực nữa!” KẾT

THÚC NẾU

KẾT THÚC



IF lồng nhau-1

BẮT ĐẦU

NHẬP NĂM Với Chúng Tôi

NHẬP bizDone

NẾU yearsWithUs ≥ 10

NẾU bizDone ≥ 5000000

HIỂN THỊ “Được phân loại là MVS”

KHÁC

HIỂN THỊ “Cần nỗ lực nhiều hơn một chút!”

KẾT THÚC NẾU

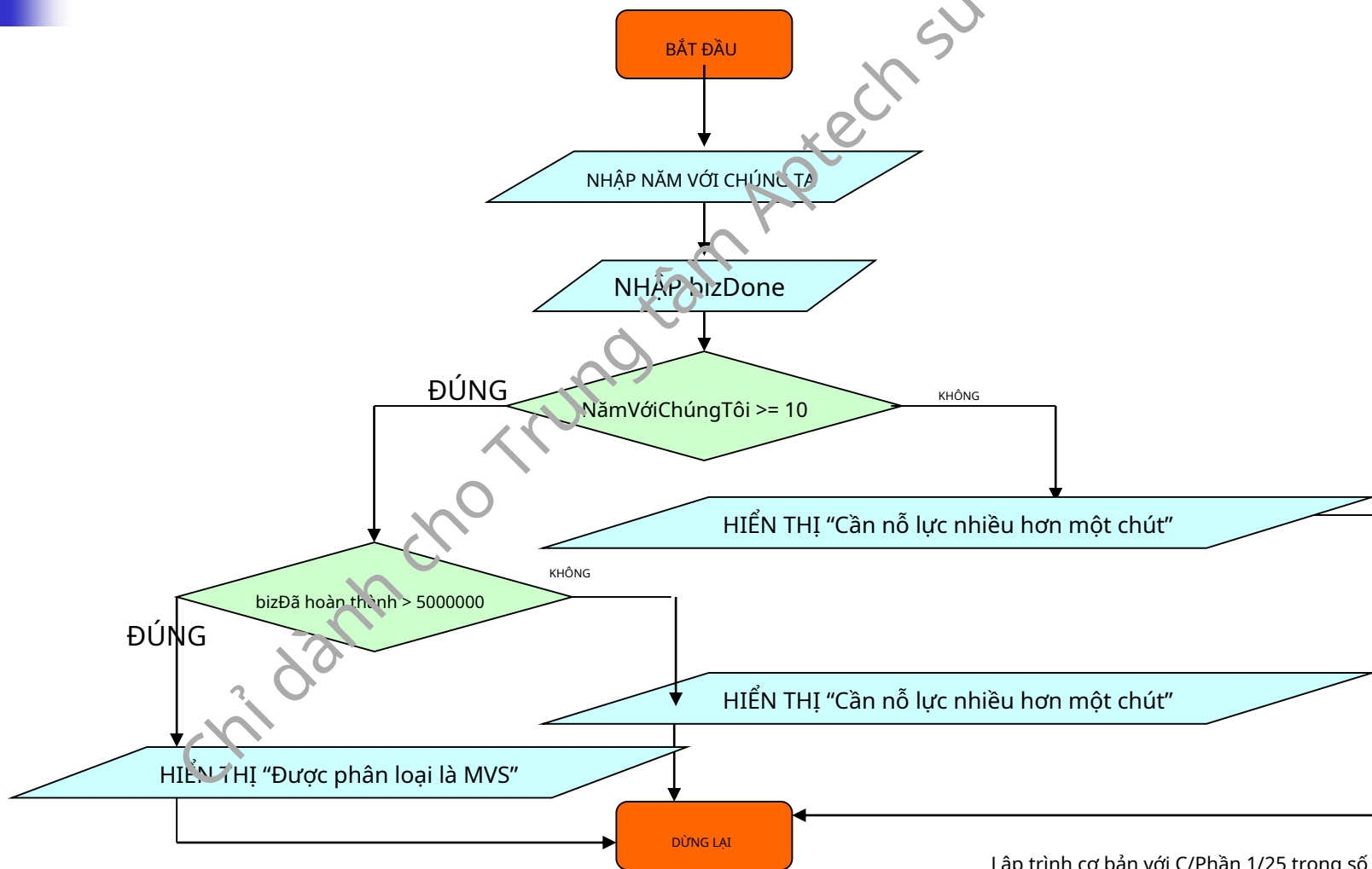
KHÁC

HIỂN THỊ “Cần thêm chút nỗ lực nữa!” KẾT

THÚC NẾU

KẾT THÚC

IF lồng nhau-2



Vòng lặp

BẮT ĐẦU

$cnt=0$

TRONG KHI($cnt < 1000$)

LÀM

HIỂN THỊ "Scooby"

$cnt=cnt+1$

KẾT THÚC

KẾT THÚC

