



TIN HỌC ĐẠI CƯƠNG

Nguyễn Duy Hiệp

hiepnd@soict.hut.edu.vn



Phần 2

Ngôn ngữ lập trình C

Tổng quan về ngôn ngữ lập trình C

Nội dung

- 2.1. Tổng quan về ngôn ngữ C
- 2.2. Kiểu dữ liệu và biểu thức
- 2.3. Các cấu trúc lập trình cơ bản
- 2.4. Mảng
- 2.5. Xâu ký tự
- 2.6. Cấu trúc - struct
- 2.7. Hàm



2.1 Tổng quan về ngôn ngữ lập trình C

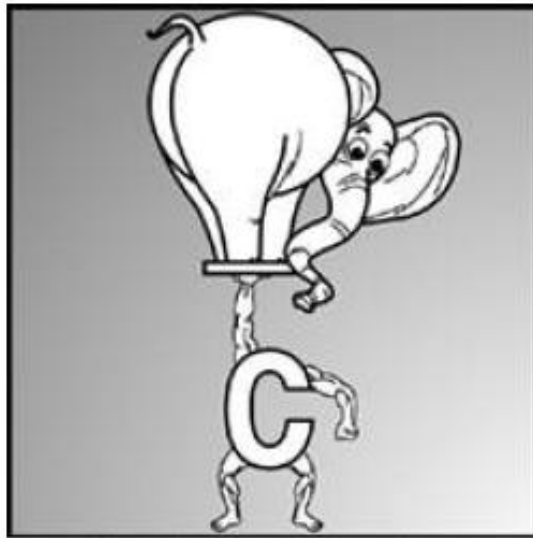
Nội dung

- 2.1.1 Tổng quan về ngôn ngữ lập trình C
- 2.1.2 Trình biên dịch C
- 2.1.3 Cấu trúc một chương trình C
- 2.1.4 Các thành phần cơ bản

2.1.1 Tổng quan về ngôn ngữ C

- Ra đời vào những năm đầu thập niên 70, do Dennis Ritchie xây dựng
- Các phiên bản C được chuẩn hóa bởi International Standard Organization (ISO): **C89, C90 và C99**
- C đã là một ngôn ngữ quan trọng và phổ biến trong quá khứ
- Cho đến ngày nay đây vẫn là một ngôn ngữ quan trọng

2.1.1 Tổng quan về ngôn ngữ C



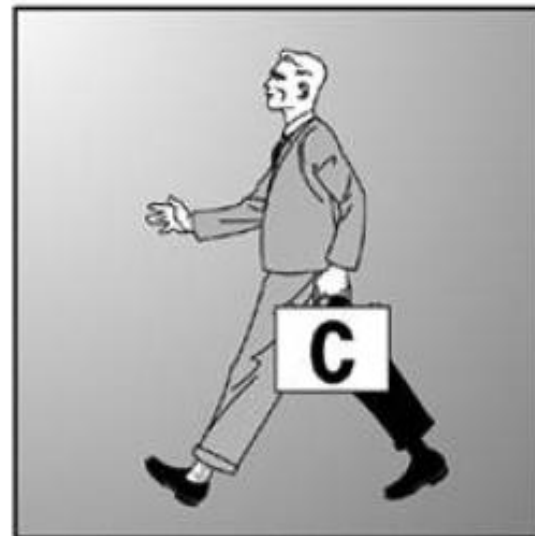
Powerful control structures



Fast



Compact code — small programs



Portable to other computers

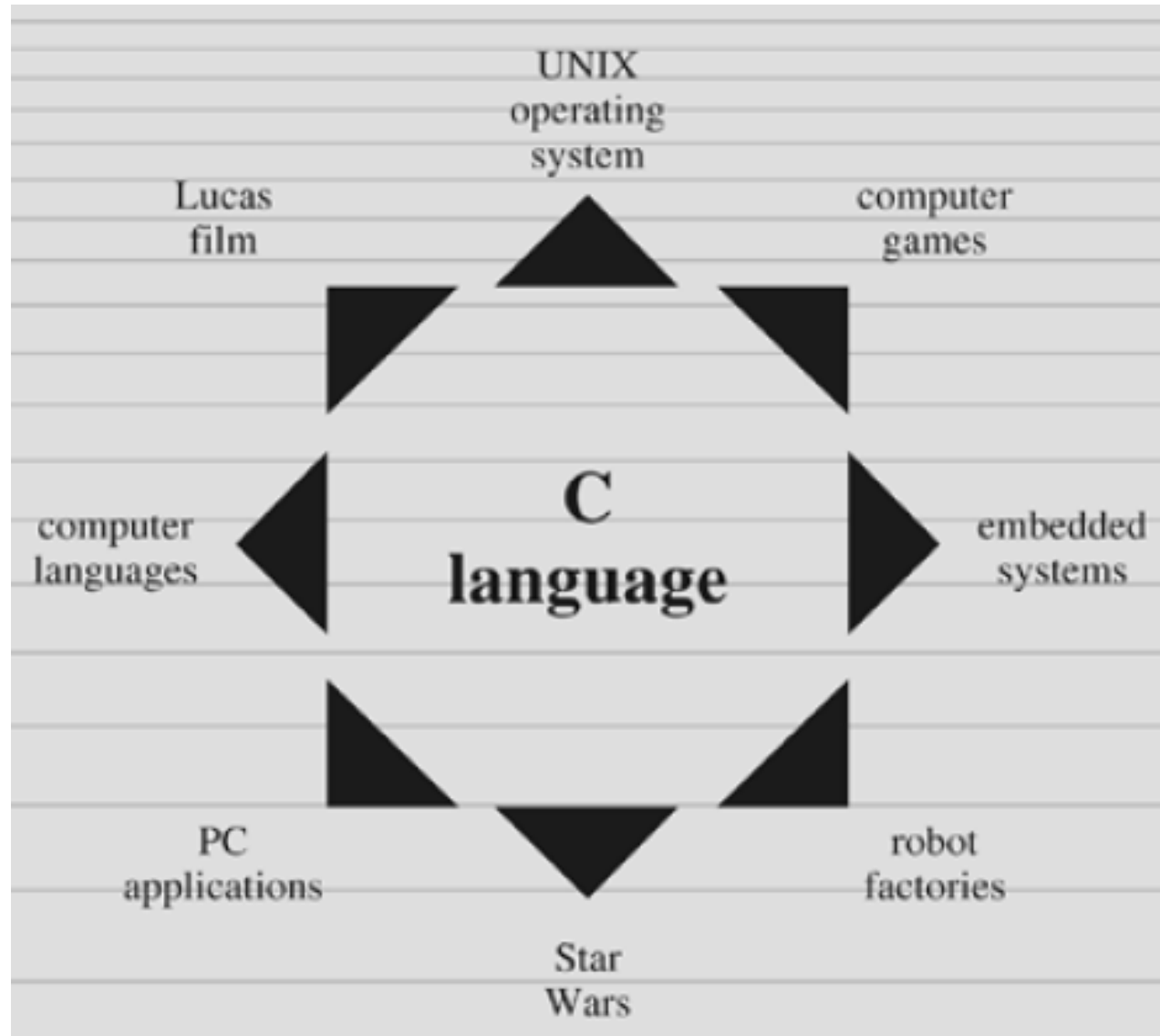
2.1.1 Tổng quan về ngôn ngữ C

- **Tính năng thiết kế:** thiết kế dựa trên lý thuyết và thực nghiệm trong khoa học máy tính. Phù hợp để thiết kế chương trình top-down, lập trình cấu trúc và thiết kế theo modular
- **Hiệu quả:** tận dụng được ưu điểm của máy tính, chương trình C nhẹ và nhanh
- **Di động, khả chuyển:** có thể thiết kế trên một hệ thống và đem sang hệ thống khác chạy mà không phải hoặc chỉ cần sửa đổi một chút
- **Mạnh và mềm dẻo:** hệ điều hành UNIX, nhiều trình dịch của các ngôn ngữ như FORTRAN, Perl, Python, Pascal, LISP, Logo, and BASIC được viết bằng C

2.1.1 Tổng quan về ngôn ngữ C

- **Hướng vào lập trình viên:** đáp ứng đầy đủ các yêu cầu của lập trình viên. Ít hạn chế so với các ngôn ngữ lập trình khác (đây là ưu điểm cũng là nhược điểm của C)
- Từ 1980, nhiều hãng phần mềm sử dụng C để xây dựng chương trình vì tính gọn nhẹ, hiệu quả cũng như khả năng di động cao.
- Ngày nay C vẫn là phần nhân của ngành công nghệ phần mềm, và là một trong 10 kỹ năng được ưa thích nhất

2.1.1 Tổng quan về ngôn ngữ C



Nội dung

- 2.1.1 Tổng quan về ngôn ngữ lập trình C
- 2.1.2 Trình biên dịch C
- 2.1.3 Cấu trúc một chương trình C
- 2.1.4 Các thành phần cơ bản

2.1.2 Trình biên dịch C

- Các bộ chương trình dịch phổ biến của ngôn ngữ C
 - Turbo C++ và Borland C++ của Borland Inc
 - MSC và VC của Microsoft Corp
 - GCC của GNU project
 - Dev-C++ của Colin Laplace
 - ...

- Nên sử dụng chương trình biên dịch C nào ?

2.1.2 Trình biên dịch C

- Cài đặt TC++ 3.0
- Cài đặt dev-C++
- Cài đặt GCC



Nội dung

- 2.1.1 Tổng quan về ngôn ngữ lập trình C
- 2.1.2 Trình biên dịch C
- 2.1.3 Cấu trúc một chương trình C
- 2.1.4 Các thành phần cơ bản

2.1.3 Cấu trúc chương trình C

Chỉ thị tiền biên dịch

```
#include <stdio.h>
#include <stdlib.h>
```

Khai báo kiểu dữ liệu
tự định nghĩa

```
typedef int km_per_hour ;
typedef int points ;
```

Khai báo hàm hoặc
nguyên mẫu hàm

```
void Display (int A[], int n);
int Sum (int a, int b) {return a+b;}
```

Khai báo biến toàn cục

```
float x,y;
```

Hàm main

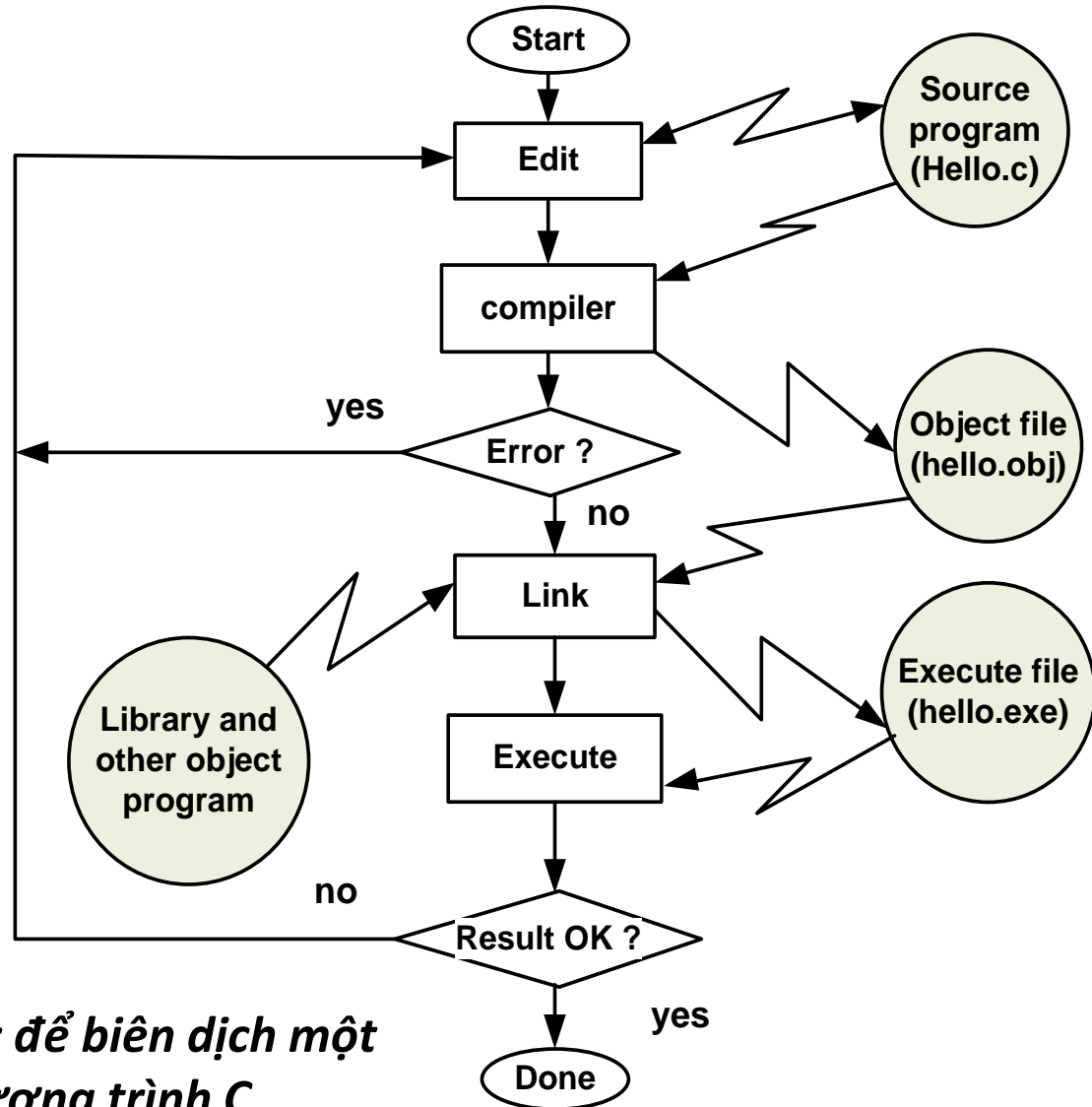
```
int main()
{ ... return 0; }
```

Khai báo thân các hàm
đã khai báo
nguyên mẫu ở trên

```
void Display (int A[], int n)
{ ... }
```

2.1.3 Cấu trúc chương trình C

Môi trường phát triển tích hợp (IDE)



Các bước để biên dịch một chương trình C

2.1.3 Cấu trúc chương trình C

- Biên dịch và chạy chương trình

```
#include <stdio.h>
int main (void)
{
    printf ("Programming is fun.\n");
    return 0;
}
```

Lưu vào file “Example_2.1.c”

2.1.3 Cấu trúc chương trình C

```
//Example_2.2.c
```

```
#include <stdio.h>
```

```
int main (void)
```

```
{
```

```
    printf ("Programming is fun.\n");
```

```
    printf ("And programming in C is even more fun.\n");
```

```
    return 0;
```

```
}
```

2.1.3 Cấu trúc chương trình C

```
//example_2.3.c
```

```
#include <stdio.h>
```

```
int main (void)
```

```
{
```

```
    printf ("Testing...\n..1\n...2\n....3\n");
```

```
    return 0;
```

```
}
```

2.1.3 Cấu trúc chương trình C

Example_2.4.c

```
#include <stdio.h>

int main (void)
{
    int value1, value2, sum;

    value1 = 50;
    value2 = 25;

    sum = value1 + value2;

    printf ("Tong cua %i va %i la %i\n", value1, value2, sum);

    return 0;
}
```

Nội dung

- 2.1.1 Tổng quan về ngôn ngữ lập trình C
- 2.1.2 Trình biên dịch C
- 2.1.3 Cấu trúc một chương trình C
- 2.1.4 Các thành phần cơ bản

2.1.4 Các thành phần cơ bản

- **Chú thích (comment):** được sử dụng để làm cho chương trình dễ đọc hơn.
- Có hai cách viết chú thích:
 - Dùng cặp `/* , */`
 - Dùng `//`

```
/* chương trình này để cộng hai số  
và hiển thị kết quả */
```

```
#include <stdio.h>
```

```
int main (void)
```

```
{
```

```
    // khai báo biến
```

```
    int value1, value2, sum;
```

```
    // gán giá trị cho 2 biến và cộng chúng lại
```

```
    value1 = 50;
```

```
    value2 = 25;
```

```
    sum = value1 + value2;
```

```
    // hiện kết quả
```

```
    printf ("Tong cua %i va %i la %i\n", value1, value2, sum);
```

```
    return 0;
```

```
}
```

2.1.4 Các thành phần cơ bản

- Biên dịch chương trình và lỗi cú pháp, lỗi logic

```
{
    ch = fgetc ( fp ) ;
    if (ch != EOF)
        printf ( "%c", ch ) ;
}
while(ch != EOF);    afaf
fclose( fp ) ;
```

[■] Message

Compiling ..\..\SAMPLE.C:
•Error ..\..\SAMPLE.C 15: Undefined symbol 'afaf'
Error ..\..\SAMPLE.C 15: Statement missing ;

F1 Help Space View source ← Edit source F10 Menu

2.1.4 Các thành phần cơ bản

■ Tập ký tự của C

26 chữ cái hoa:	A B C ... X Y Z
26 chữ cái thường:	a b c ... x y z.
10 chữ số:	0 1 2 3 4 5 6 7 8 9.
Các kí hiệu toán học:	+ - * / = < >
Các dấu ngăn cách:	. ; , : space tab
Các dấu ngoặc:	() [] { }
Các kí hiệu đặc biệt:	- ? \$ & # ^ \ ! \ " ~ .v.v.

2.1.4 Các thành phần cơ bản

- **Từ khóa (Keyword)** là những từ có sẵn của ngôn ngữ và được sử dụng dành riêng

break	case	char	const	continue	default
do	double	else	enum	float	for
goto	if	int	interrupt	long	return
short	signed	sizeof	static	struct	switch
typedef	union	unsigned	void	while	

- **Chú ý:** Tất cả các từ khóa trong C đều viết bằng chữ thường.

2.1.4 Các thành phần cơ bản

- Các từ khóa trong C được sử dụng để
 - Đặt tên cho các kiểu dữ liệu: **int, float, double, char, struct, union...**
 - Mô tả các lệnh, các cấu trúc điều khiển: **for, do, while, switch, case, if, else, break, continue...**

