

KỸ THUẬT LẬP TRÌNH C/C++

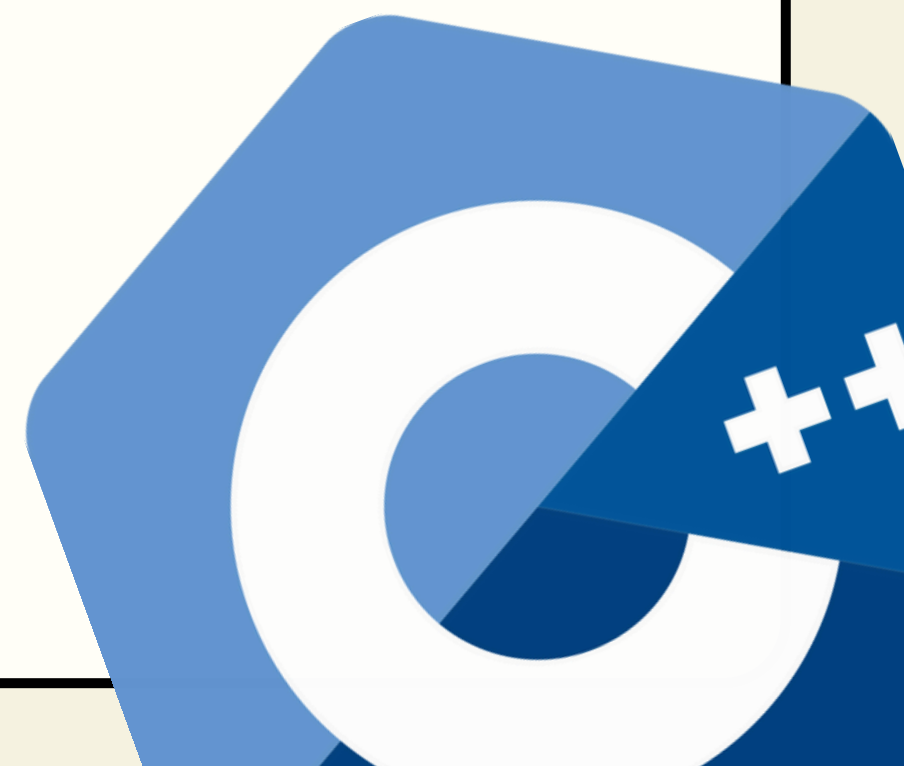
Dạy bởi 1 SVBK

Đại học Bách Khoa Hà Nội
Kỹ thuật Điện tử - Viễn thông

2024-25

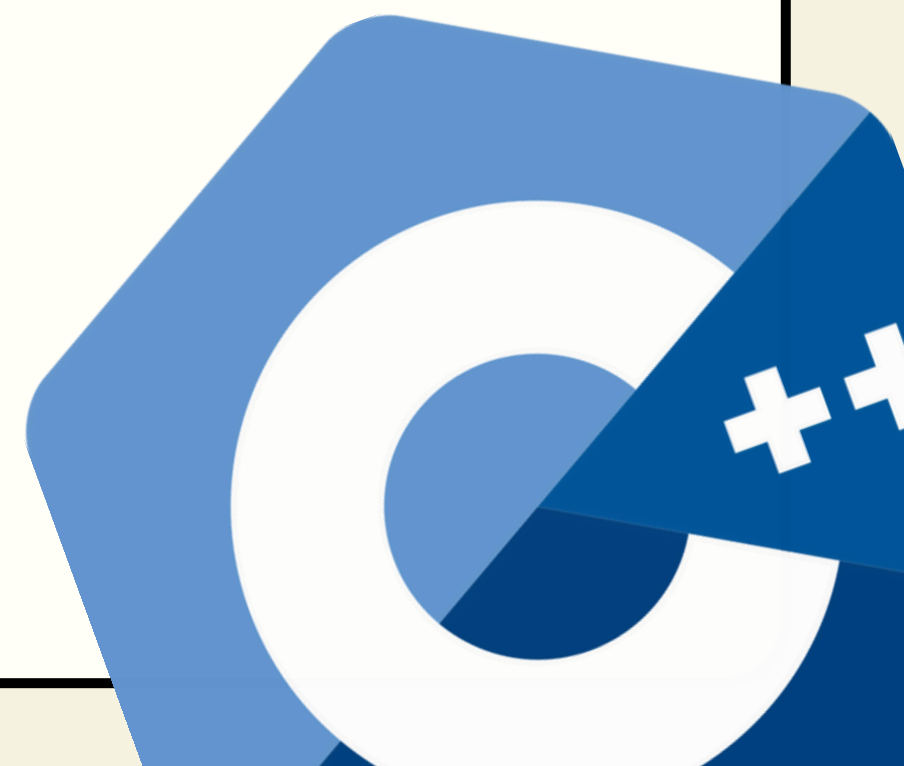
THÔNG TIN MÔN HỌC

- Kỹ thuật lập trình & ngôn ngữ C/C++
- Giới thiệu qua về lập trình & ngôn ngữ lập trình
- Ngôn ngữ C/C++
- Bài giảng: GG Drive
- Cách thức làm việc: Messenger & Discord, Dev C++.



NỘI DUNG MÔN HỌC

- **Chương I:** Cơ bản về lập trình. Giới thiệu về các ngôn ngữ lập trình.
- **Chương II:** Ngôn ngữ C
- **Chương III:** Ngôn ngữ C++



GIỚI THIỆU VỀ LẬP TRÌNH

Lập trình là việc sử dụng các **ngôn ngữ lập trình**, các **đoạn mã lệnh** và các tiện ích có sẵn để xây dựng các chương trình phần mềm, trò chơi ứng dụng của hệ thống xử lý các trang web... giúp người dùng có thể thực hiện các mệnh lệnh của máy tính cũng như tương tác qua lại với nhau thông qua các thiết bị điện tử.

Lập trình chỉ là 1 nhánh trong CNTT, không phải là CNTT.

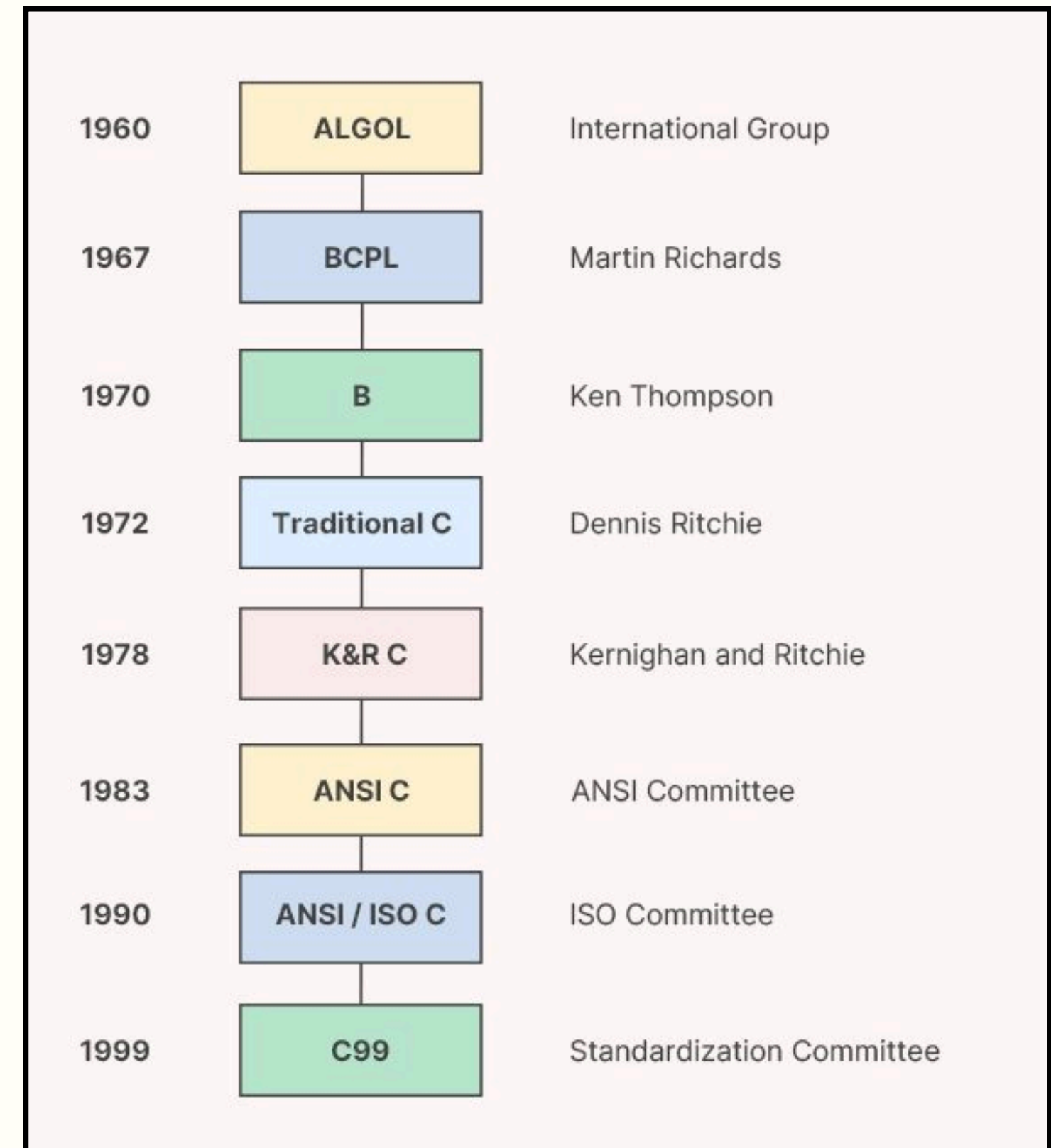
TỔNG QUAN VỀ NGÔN NGỮ C

- Ngôn ngữ lập trình C là một ngôn ngữ lập trình bậc cao, đa chức năng (general-purpose), cung cấp giao diện trực tiếp, nhất quán và mạnh mẽ cho các hệ thống lập trình. Đây là lý do vì sao ngôn ngữ C được áp dụng rộng rãi trong việc phát triển phần mềm hệ thống, phần mềm ứng dụng và các hệ thống nhúng.
- Sự linh hoạt và mạnh mẽ của ngôn ngữ C khiến nó được gọi là “mẹ của mọi ngôn ngữ”. Nhiều ngôn ngữ lập trình khác được phát triển dựa trên C như **C++**, **PHP**, **JavaScript**, **Java**, **Python**,...
- Bởi vì là một ngôn ngữ có tính chất general-purpose, C có khả năng thích ứng tốt với phát triển hệ thống, ví dụ như hệ điều hành, trình biên dịch và trình điều khiển mạng. Tuy nhiên một số người cho rằng cú pháp của C có thể phức tạp và khó học, trong khi những người khác nhấn mạnh vấn đề thiếu chuẩn hóa.

TỔNG QUAN VỀ NGÔN NGỮ C

LỊCH SỬ PHÁT TRIỂN

- Kế thừa **ngôn ngữ lập trình B**.
- Ngôn ngữ C được phát triển bởi Dennis Ritchie vào năm 1972 tại Bell Labs.
- Năm 1983: ANSI (American National Standards Institute), một nhóm được thành lập để tiêu chuẩn hóa ngôn ngữ C, làm việc và xuất bản ngôn ngữ C vào 1989.

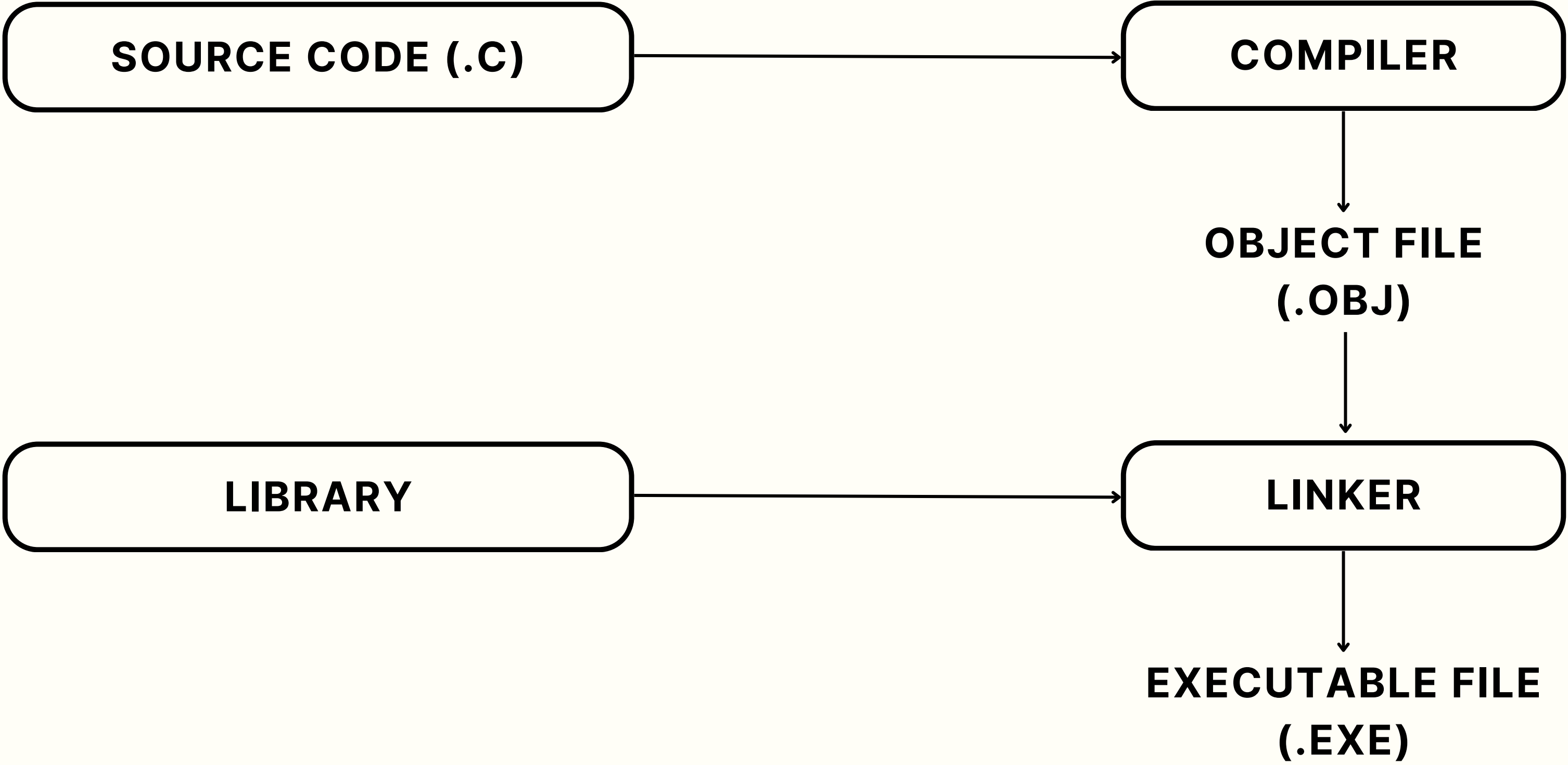


TỔNG QUAN VỀ NGÔN NGỮ C

CÁC ĐẶC ĐIỂM CHÍNH CỦA NGÔN NGỮ C

- **Đa mục đích** (General-purpose)
- **Tính di động** (Portability)
- **Hỗ trợ lập trình cấu trúc** (Structured Programming)
- **Truy cập cấp thấp** (Low-level access)
- **Hệ thống kiểu mạnh mẽ** (Strong typing system)
- **Thư viện chuẩn phong phú** (Rich Standard Library)
- **Hỗ trợ con trỏ** (Pointer Support)
- **Tối ưu hóa hiệu suất** (Performance Optimization)
- **Khả năng mở rộng** (Extendibility)

CÁCH NGÔN NGỮ C HOẠT ĐỘNG



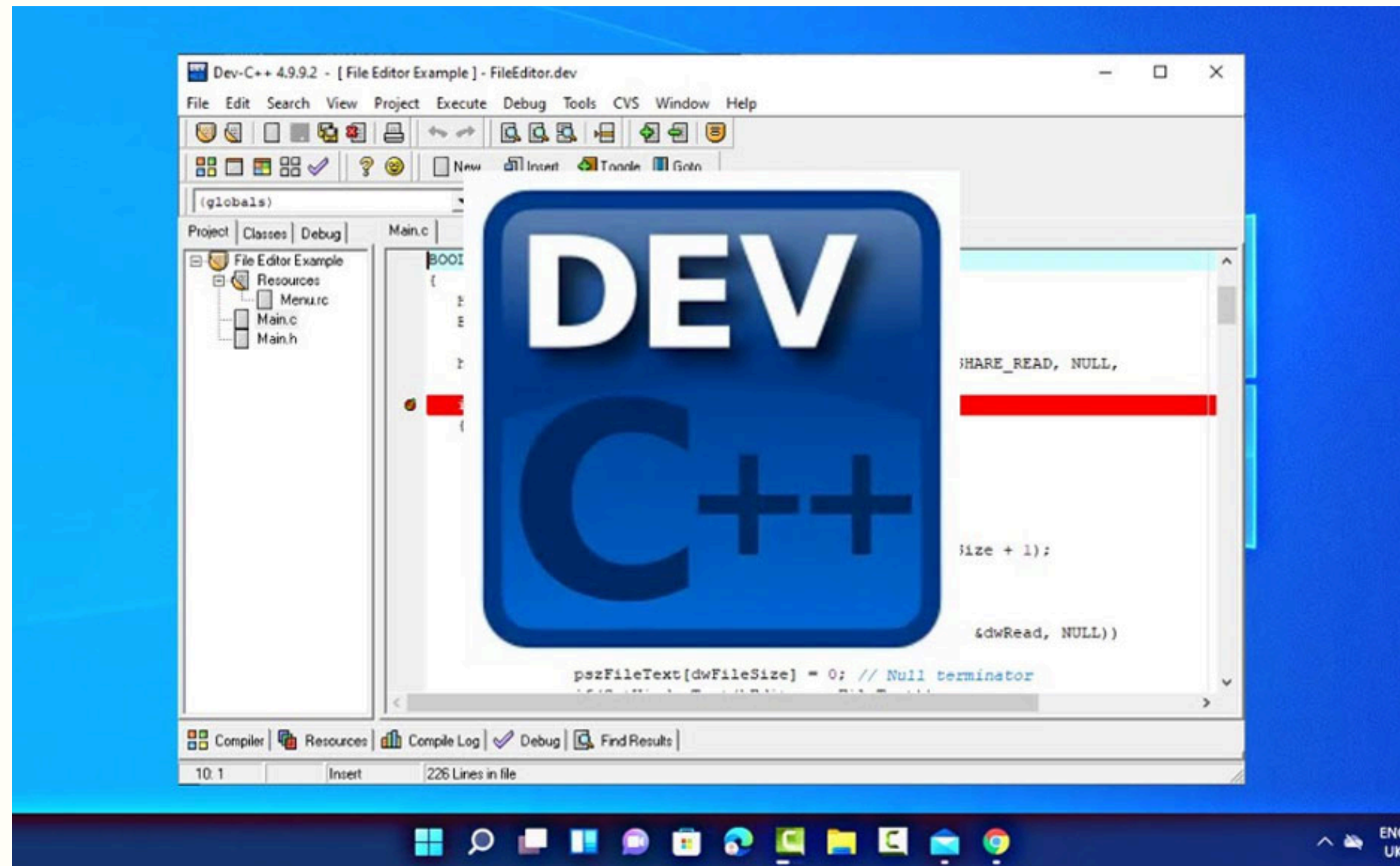
ỨNG DỤNG NGÔN NGỮ C

- Với những ưu điểm đầy nổi bật như trên, hiện tại ngôn ngữ lập trình C được sử dụng vô cùng phổ biến tại nhiều ứng dụng, phần mềm, chương trình, thiết bị,... khác nhau tại smartphone, laptop, máy tính, robot, website & các hệ điều hành. Những lĩnh vực nổi bật như: **Thiết bị điện tử, công nghệ điện tử và thiết bị gia dụng đều ứng dụng ngôn ngữ C để tạo ra các phần mềm điều khiển.**
- Hiệu quả xử lý nhanh chóng đầy đa dạng của ngôn ngữ lập trình C đã giúp ích không nhỏ trong việc theo dõi, phát triển hệ thống quản lý nhân sự. Cũng như **điều khiển & chế ngự máy, website, ứng dụng, phần mềm,... cực kỳ tốt.**

CÔNG CỤ SỬ DỤNG

- **Visual Studio:** Visual Studio là một Môi trường Phát triển Tích hợp (IDE) dành cho Windows, được phát triển bởi Microsoft. Nó bao gồm trình biên dịch tích hợp riêng, được gọi là Microsoft Visual C++ (MSVC).
- **Visual Studio Code (VSCode):** Visual Studio Code là một trình soạn thảo mã nguồn mở, miễn phí và nhẹ, được phát triển bởi Microsoft. Nó có thư viện tiện ích mở rộng phong phú giúp nâng cao chức năng phát triển C++.
- **Sublime Text:** Sublime Text là một trình soạn thảo mã nguồn đa nền tảng, phổ biến nhờ tốc độ nhanh và thiết kế tối giản. Nó hỗ trợ lập trình C++ thông qua các plugin và có nhiều chủ đề, gói mở rộng để tùy chỉnh.
- **Dev-C++:** Dev-C++ là một môi trường phát triển tích hợp (IDE) miễn phí và mã nguồn mở dành cho lập trình C và C++. Nó sử dụng trình biên dịch MinGW (Minimalist GNU for Windows) để biên dịch mã nguồn. Dễ sử dụng cho người mới học.

CÔNG CỤ SỬ DỤNG



CHƯƠNG TRÌNH C ĐẦU TIÊN

Bao gồm các hàm đầu vào tiêu chuẩn đầu vào. Hàm printf() được định nghĩa trong stdio.h.

```
1  #include<stdio.h>
2
3  int main() {
4      printf("Hello World");
5      return 0;
6  }
```

Chức năng chính được gọi đầu tiên - Hàm main. Code trong {} sẽ được thực thi.

Hàm printf() (hàm có trong thư viện): in dữ liệu ra màn hình.

Trả về 0 khi chương trình hoạt động chính xác. Kết thúc chương trình. Nếu có lệnh nào khác dưới câu lệnh này sẽ không được thực thi nữa.

CÚ PHÁP (SYNTAX) CỦA C

```
1 #include<stdio.h>
2
3 int main() {printf("Hello World");return 0;}
```

Chương trình cũng có thể được viết như sau:

=> C không quan tâm đến các khoảng trắng (blank space)

Tất cả các câu lệnh trong C đều kết thúc bằng dấu chấm phẩy (semicolon).

Khi làm việc với các văn bản (text) đều phải sử dụng dấu ngoặc kép (double quotes).

CÚ PHÁP (SYNTAX) CỦA C

```
1 #include <stdio.h>
2
3 int main () {
4     printf("Toi la svbk \n");
5     printf("I have no life");
6     return 0;
7 }
```

Có \n

Không có \n




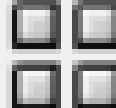
Muốn in nhiều câu khác nhau có thể dùng nhiều câu lệnh printf hoặc chung 1 câu lệnh. Nhưng để xuống dòng thì phải có ký tự “**endline**” - \n

```
Toi la svbk
I have no life
-----
Process exited after 0.04351 seconds with return value 0
Press any key to continue . . . _
```

```
Toi la svbkI have no life
-----
Process exited after 0.04966 seconds with return value 0
Press any key to continue . . .
```

CÁCH CHẠY CHƯƠNG TRÌNH C

Trong mục Execute có 4 phần sau:

	Compile	F9
	Run	F10
	Compile & Run	F11
	Rebuild All	F12

Compile: Ngôn ngữ C là ngôn ngữ để con người hiểu, cần phải **compile - biên dịch** sang ngôn ngữ máy để **run - chạy**.

Cứ F11 mà phang nhé.

CHÚ THÍCH (COMMENT) TRONG C

```
1 // Đây là 1 comment
2
3 /*
4  Ho va ten: Colleen Hoang
5  Truong: Doi`
6  */
7 #include <stdio.h>
8
9 int main () {
10     printf("Toi thích học ngôn ngữ C");
11     return 0;
12 }
```

Comment 1 dòng bắt đầu bởi //

Comment nhiều dòng bắt đầu bởi /* và đóng bởi */

C sẽ bỏ qua tất cả những thứ thuộc về comment.

BIẾN (VARIABLES) TRONG C

- Một biến trong C là tên của vị trí bộ nhớ. Nó được sử dụng để lưu trữ dữ liệu. Giá trị của nó có thể được thay đổi và nó có thể được sử dụng lại nhiều lần. Mỗi biến trong C có một loại dữ liệu cụ thể, xác định kích thước của bộ nhớ của biến; phạm vi các giá trị có thể được lưu trữ trong bộ nhớ đó.
- Biến là một cách để **thể hiện vị trí bộ nhớ thông qua một cái tên để nó có thể được xác định một cách dễ dàng**. Tên của một biến có thể **bao gồm các chữ cái, chữ số và ký tự gạch dưới**. Nó phải bắt đầu bằng một chữ cái thường hoặc một gạch dưới. Biến trong C có **phân biệt chữ hoa và chữ thường**.

BIẾN (VARIABLES) TRONG C

Chú ý về đặt tên cho biến:

Trong lập trình quy chuẩn đầu tiên cần đưa ra là quy chuẩn cho việc đặt tên.

Có 3 quy tắc cần chú ý:

- **underscore:** sử dụng dấu gạch chân giữa các từ, tất cả các từ đều viết thường, ví dụ: `this_is_my_variable`.
- **camelCase (hay được sử dụng nhất):** giống như cách viết của nó, từ đầu tiên viết thường, các từ tiếp theo viết hoa chữ cái đầu, ví dụ `$thisIsMyVariable`.
- **PascalCase:** viết hoa tất cả các chữ cái đầu, ví dụ `$ThisIsMyVariable`.

BIẾN (VARIABLES) TRONG C

Và các quy tắc chung sau cần chú ý:

- Tên có thể chứa chữ cái, chữ số và dấu gạch dưới (_).
- Tên phải bắt đầu bằng một chữ cái hoặc dấu gạch dưới (_).
- Tên có phân biệt chữ hoa và chữ thường (ví dụ: myVar và myvar là hai biến khác nhau).
- Tên không được chứa khoảng trắng hoặc các ký tự đặc biệt như !, #, %, v.v.
- Không được sử dụng các **từ khóa** dành riêng (reserved words) như int làm tên biến.
- Đặt tên nên đặt cho dễ hiểu, x hay y nghe gọn đấy nhưng nên đặt dễ hiểu nó là biến cho cái gì như age, studentName,...

BIẾN (VARIABLES) TRONG C

Trong C có nhiều kiểu dữ liệu khác nhau, khi khai báo sẽ dùng các **từ khóa**. Đây là 3 loại cơ bản:

- int – lưu trữ số nguyên (không có phần thập phân), ví dụ: 123 hoặc -123.
- float – lưu trữ số thực (số có phần thập phân), ví dụ: 19.99 hoặc -19.99.
- char – lưu trữ một ký tự đơn, ví dụ: 'a' hoặc 'B'. Ký tự được đặt trong dấu nháy đơn (').

Đây sẽ là 3 kiểu dữ liệu sẽ tạm thời sử dụng khi đang tìm hiểu về biến, chúng ta sẽ tìm hiểu kĩ hơn về các kiểu dữ liệu khác trong phần sau.

BIẾN (VARIABLES) TRONG C

CÚ PHÁP KHAI BÁO BIẾN

```
kiểuDữLiệu tênBiến = giáTrị;
```

hoặc

```
kiểuDữLiệu tênBiến;
```

(không phải lúc nào cũng cần giá trị luôn)

VÍ DỤ VỀ CÁC KHAI BÁO HỢP LỆ:

```
int myAge = 20;  
float funny = 6.9;  
char c = 'A';  
int a = 100, b, c;
```

VÍ DỤ VỀ CÁC KHAI BÁO KHÔNG HỢP LỆ:

```
int float;  
float tuoi HS = 20;  
char #hay = 'c';
```

BIẾN (VARIABLES) TRONG C

ĐỊNH DẠNG ĐẶC TẢ (FORMAT SPECIFIER) & IN BIẾN RA MÀN HÌNH

Các định dạng đặc tả (format specifiers) được sử dụng cùng với hàm `printf()` để cho trình biên dịch biết kiểu dữ liệu mà biến đang lưu trữ. Chúng hoạt động như một chỗ giữ chỗ (placeholder) cho giá trị của biến.

Một định dạng đặc tả bắt đầu bằng dấu phần trăm %, theo sau là một ký tự xác định kiểu dữ liệu. Ví dụ, để hiển thị giá trị của một biến kiểu **int**, ta sử dụng **định dạng đặc tả %d, đặt bên trong dấu ngoặc kép (")**, cuối cùng là **tên biến ngăn cách bằng dấu phẩy** trong hàm `printf()`:

```
int myNumber = 25;
printf("Giá trị của biến là: %d", myNumber); // Output: 25;
```

Ta sử dụng **%c** cho char và **%f** cho float.

BIẾN (VARIABLES) TRONG C

VÍ DỤ

```
1  #include <stdio.h>
2
3  int main() {
4      int myNum = 15;
5      float myFloatNum = 5.99;
6      char myLetter = 'D';
7      printf("%d\n", myNum);
8      printf("%f\n", myFloatNum);
9      printf("%c\n", myLetter);
10     return 0;
11 }
```

```
15
5.990000
D
-----
Process exited after 0.04291 seconds
Press any key to continue . . .
```

Kiểu dữ liệu (DATATYPES) TRONG C

- Kiểu dữ liệu là phần xác định các giá trị mà **một biến có thể nhận** hay **giá trị mà một hàm có thể trả về**. Các kiểu dữ liệu này đã được lưu trữ trong chương trình C.
- 4 kiểu dữ liệu cơ bản:
 - **Kiểu dữ liệu cơ bản:** Chúng là các loại số học và được phân loại thành: các kiểu số nguyên và các kiểu số thực.
 - **Kiểu Enumeration:** Chúng lại là các loại số học và chúng được sử dụng để xác định các biến chỉ có thể chỉ định một số giá trị số nguyên rời rạc trong suốt chương trình.
 - **Kiểu dữ liệu Derived:** Chúng bao gồm các kiểu con trỏ (Pointer), mảng (Array), cấu trúc (Structure), Union và hàm (function).
 - **Kiểu Void:** Kiểu Void (vô nghĩa) chỉ định rằng không có giá trị nào có sẵn.

Kiểu dữ liệu (DATATYPES) TRONG C

Kiểu dữ liệu Enum (thường gặp trong bài kiểm tra)

Là kiểu dữ liệu liệt kê - enumeration.
Để tạo một enum trong C, dùng từ khóa enum, theo sau là tên của enum và tách riêng các mục enum bằng dấu phẩy:

```
enum Level {LOW, MEDIUM, HIGH};  
  
int main() {  
    enum Level myVar = MEDIUM;  
    printf("%d", myVar);  
}
```

Output của đoạn code đó bằng 1. Vì như mặc định, nếu không cho trước giá trị khi khai báo enum thì, phần tử đầu tiên sẽ giá trị là 0, và tăng dần theo trái sang phải. Nếu cho phần tử thứ nhất là x thì phần tử thứ 2 (nếu chưa có giá trị khi khai báo) sẽ có giá trị là x+1. Tương tự đối với các phần tử về sau. Đoạn code này sẽ in ra kết quả là 21:

```
enum Level {LOW = 20, MEDIUM, HIGH = 90};  
  
int main() {  
    enum Level myVar = MEDIUM;  
    enum Level myVar1 = HIGH;  
    printf("%d", myVar);  
}
```

Kiểu dữ liệu (DATATYPES) TRONG C

Các kiểu **dữ liệu cơ bản** & đặc tả của kiểu dữ liệu đó:

Kiểu dữ liệu	Kích thước (byte)	Giá trị có thể lưu	Giới hạn có thể lưu	Đặc tả (Format Specifier)
short	2 byte	Số nguyên	-32,768 -> 32,767	%hi
unsigned short	2 byte	Số nguyên	0 -> 65,535	%hu
int	4 byte	Số nguyên	-2,147,483,648 -> 2,147,483,647	%d
unsigned int	4 byte	Số nguyên	0 -> 4,294,967,295	%u
long long	8 byte	Số nguyên	-9223372036854775808 -> 9223372036854775807	%lld
unsigned long long	8 byte	Số nguyên	0 -> 18446744073709551615	%llu
char	1 byte	Số nguyên, Ký tự	-128 -> 127	%c
unsigned char	1 byte	Số nguyên, Ký tự	0 -> 255	%c
float	4 byte	Số thực	3.4E-38 -> 3.4E+38	%f
double	8 byte	Số thực	1.7E-308 -> 1.7E+308	%lf

KIỂU DỮ LIỆU (DATATYPES) TRONG C

5 cái quan trọng nhất & thường dùng:

- **Số nguyên:** int (%d) và long long (%lld) (số to)
- **Số thực:** float (%f) và double (%lf) (ưu tiên)
- **Ký tự:** char (%c) (chú ý: giá trị của char dưới dạng thập phân sẽ theo bảng ASCII).

NHẬP XUẤT DỮ LIỆU TRONG C

XUẤT DỮ LIỆU

- **Chuỗi định dạng** có thể là nội dung của một chuỗi ký tự hoặc đặc tả kiểu dữ liệu của biến mà ta muốn in ra màn hình.
- **Đối số**: Đây thường là các biến mà ta sẽ in ra, số lượng đối số sẽ bằng số lượng đặc tả trong chuỗi định dạng.

CÚ PHÁP

`printf("Chuỗi định dạng", Đối số);`

```
#include <stdio.h>

int main () {
    int a = 20;
    printf("Tuoi cua tao la: %d tuoi", a);
    return 0;
}
```

`Tuoi cua tao la: 20 tuoi`

NHẬP XUẤT DỮ LIỆU TRONG C

XUẤT DỮ LIỆU

Chú ý: đối với **float** và **double**, khi in ra giá trị của float và double thì đề bài thường yêu cầu các bạn in ra kết quả với 1 vài chữ số phần thập phân. Đặc tả của float là %f và của double là %lf, cú pháp để in ra x chữ số sau dấu phẩy đối với float và double ta sử dụng đặc tả %.xf hoặc %.xlf. (Máy tính sẽ tự làm tròn số.)

VÍ DỤ:

In ra số thực với độ chính xác 2 chữ số thập phân:

```
2
3 int main () {
4     float x = 10.3789423798427389;
5     printf("Ket qua: %.2f", x);
6     return 0;
7 }
```

Ket qua: 10.38

NHẬP XUẤT DỮ LIỆU TRONG C

NHẬP DỮ LIỆU

Khi nhập giá trị cho 1 biến từ bàn phím ta cần truyền đặc tả của nó vào phần chuỗi định dạng, kèm theo dấu & trước tên biến. Dấu & này thể hiện địa chỉ của biến trong bộ nhớ, có thể hiểu đơn giản mỗi khi nhập giá trị cho biến từ bàn phím thì hàm scanf sẽ tìm đến địa chỉ của biến đó trong bộ nhớ để **gán** cho nó giá trị mà bạn đã nhập từ bàn phím.

CÚ PHÁP

```
scanf("Đặc tả", &biến);
```

```
int main () {  
    int n;  
    printf("Nhap n: ");  
    scanf("%d", &n);  
    printf("Gia tri cua n la: %d", n);  
    return 0;  
}
```

```
Nhap n: 100  
Gia tri cua n la: 100  
-----
```

NHẬP XUẤT DỮ LIỆU TRONG C

NHẬP DỮ LIỆU

Chú ý với **char**: Để nhập 1 ký tự từ bàn phím có thể dùng hàm scanf hoặc getchar().

CÚ PHÁP

biến = getchar();

```
int main(){
    char kitu;
    printf("Nhap ki tu: ");
    kitu = getchar();
    printf("Ki tu vua nhap: %c\n", kitu);
    return 0;
}
```

```
Nhap ki tu: a
Ki tu vua nhap: a
-----
```