***FUNCTION***

***(Hàm)***

**I.Định nghĩa**

- Hàm là 1 đoạn chương trình được đóng gói thành đơn vị thực thi dung để thực hiện 1 tác vụ độc lập trong chương trình.

- Hàm còn được gọi là chương trình con

- Lợi ích của hàm:

+ Cho phép tái sử dụng mã nguồn, hàm chỉ cần định nghĩa 1 lần nhưng có thể sử dụng nhiều lần.

+ Sử dụng hàm viết chương trình dễ phát triển, dễ gỡ lỗi, dễ bảo trì.

Có 2 loại hàm:

* Hàm dựng sẵn: Hàm đc cung cấp bởi C++
* Hàm do ng dung tự định nghĩa: Tự viết ra hàm

**Chú ý: Để sử dụng những hàm có sẵn trong thư viện C++ ta cần phải #include <thư viện> trc khi sử dụng**

* Lệnh **Return** dùng để thoát ra khỏi hàm và trả về giá trị của hàm
* Hàm có thể có các những tham số, Tham số dung để nhận giá trị từ bên ngoài truyền vào hàm
* Tham số là các biến đc khai báo trong đầu hàm, tham số giúp thay đổi kết quả/nội dung của hàm
* Khi định nghĩa hàm ta cần khai báo kiểu trả về của hàm, nếu hàm k trả về giá trị thì kiểu trả về sẽ là kiểu **void**

**Chú ý: Khi định nghĩa hàm có bao nhiêu tham số thì khi gọi hàm ta cần truyền báo bấy nhiêu giá trị tương ứng.**

* Khi gọi hàm có thể gán kết quả của hàm cho 1 biến nào đó để xử lí
* Đối số (Argument) : Là những biến hay giá trị đc truyền từ bên ngoài vào trong hàm
* Function prototype ( Nguyên mẫu hàm ): Nếu muốn gọi hàm trc định nghĩa hàm ta cần khai báo tên hàm ở đầu chương trình, lệnh khai báo gọi là Nguyên mẫu hàm
* Multiverse Function/ Function Overloading ( Nạp chồng ): là quá trình định nghĩa 2 hoặc nhiều hàm có cùng tên nạp chồng hàm nhưng có danh sách tham số khác nhau. Danh sách tham số khác nhau ở 2 tiêu chí:

+ Số lượng tham số

+ Kiểu dữ liệu của tham số

* Tham số có giá trị mặc định là các tham số khi đc khai báo sẽ đc gán giá trị mặc định có tác dụng tránh lỗi khi ta quên k truyền đối số.
* Nếu ta k truyền đối số thì tham số sẽ đc gán giá trị mặc định. Nếu ta có truyền đối số thì tham số sẽ lấy giá trị đc truyền

**Chú ý: Khi định nghĩa hàm có tham số mặc định các tham số tiếp theo cx phải khai báo mặc định**

Kiểu dữ liệu của biến:

* Dùng để xác định loại dữ liệu mà biến đó có thể lưu trữ
* Lớp lưu trữ là 1 đặc điểm của biến cho phép xác định phạm vi và thời gian tồn tại của biến đó
* Phạm vi của biến là đoạn chương trình mà biến đó có thể lưu trứ
* Tgian tồn tại của biến là khoảng thời gian biến đấy còn tồn tại
* Biến cục bộ ( Local Variable ): là 1 biến đc khai báo ở trong 1 hàm, phạm vi truy cập của biến cục bộ chỉ trong 1 hàm đó, biến cục bộ chỉ tồn tại chừng nào hàm còn đang hoạt động. Hàm kết thúc biến cục bộ sẽ bị hủy
* Biến toàn cục ( Global Variable ): Là biến đc khai báo ở bên ngoài các hàm. Phạm vi biến toàn cục la tất cả các hàm trong chương trình, là toàn chương trình. Biến toàn cục sẽ tồn tại chừng nào chương trình hoạt động. Biến toàn cục đc sử dụng để lưu trữ các giá trị dùng cho toàn bộ chương trình
* Từ khóa static dùng để khai báo 1 biến tĩnh
* Biến tĩnh có phạm vi truy cầm chỉ trong hàm mà nó khai báo, tgian tồn tại của biến tĩnh cho đến khi chương trình kết thúc
* Biến thanh ghi ( Register variable ): chỉ truy cập đc trong nội bộ của hàm
* Biến “thread”: Dùng cho những ứng dụng đa luồng