

CONFIDENTIAL

C Programming Introduction

week 9: Function

For HEDSPI Project

Hàm

- Chứa 1 tập các khai báo và câu lệnh, được đặt tên gợi nhớ nội dung thực hiện
- Hàm thường trả về 1 giá trị
- Là 1 chương trình con
 - Khi viết chương trình, ta luôn viết hàm **main**
 - Trong **main** có thể gọi đến các hàm
 - Bản thân các hàm có thể gọi đến các hàm khác nữa

Ví dụ: Square

```
double square(double a)
{
    return a * a;
}
```

This is a function defined outside main

```
int main(void)
{
    double num = 0.0, sqr = 0.0;

    printf("enter a number\n");
    scanf("%lf", &num);

    sqr = square(num);
    printf("square of %g is %g\n", num, sqr);

    return 0;
}
```

Here is where we call the function square

Tại sao dùng hàm?

- Chia chương trình thành các công việc nhỏ hơn (chia để trị)
 - Giải quyết các bài toán lớn theo cách này dễ hơn
- Có thể có 1 tập các câu lệnh được viết lặp đi lặp lại
 - Sử dụng hàm tránh được việc viết lại nhiều lần
- Sử dụng hàm làm cho chương trình dễ đọc và quản lý hơn

Đặc điểm của hàm

<kiểu trả về> name(<ds tham số>)

```
{
    khai báo biến cục bộ
    các câu lệnh
    return <giá trị trả về>;
}
```

- Khi dùng lời gọi hàm, ta có thể đưa các tham số của hàm vào <ds tham số>

Bài 9.1

- Viết hàm tính giá trị của biểu thức trọng trường
 $ke = mv^2/2$, với m là trọng lượng (kg) và v là tốc độ (m/s)
- Dùng hàm này trong chương trình

Bài 9.2

1. Viết hàm `is_prime` nhận đầu vào là 1 số nguyên dương, trả về 1 nếu nó là số nguyên tố, 0 nếu ngược lại
khai báo: `int is_prime(int n);`
2. Viết chương trình nhập số nguyên dương `N`, in ra tất cả các số nguyên tố từ 2 đến `N`.

Tính hàm từ (1)!

Truyền giá trị

- Các giá trị tham số của hàm được sao chép và sử dụng, không thao tác trực tiếp trên các biến thực sự được truyền vào.
- Mọi thay đổi giá trị tham số trong thân hàm không thay đổi giá trị biến trong lời gọi hàm.

Bài 9.3

- Viết chương trình cài đặt các hàm sau. Sử dụng chúng trong chương trình chính.
 - 1 hàm tính tổng lập phương các số nguyên từ 1 đến `n`
 - 1 hàm in ra tất cả các ước số của `n`
 - 1 hàm in ra bình phương của các số từ 1 đến `n`.

Bài 9.4

- Viết chương trình tính lương của công nhân theo tuần. Lương trung bình là 15000vnd/h. Mỗi công nhân phải làm 40h/tuần. Nếu họ làm việc quá giờ, tiền được trả gấp 1.5 lần so với mức bình thường.
- Kiểm tra dữ liệu: số giờ làm việc của 1 công nhân $\geq 10h$, $\leq 65h$ 1 tuần.

Bài 9.5

- Công thức chuyển đổi nhiệt độ Fahrenheit sang Celcius là $C = 5/9(F-32)$
- Viết hàm `celsius` với tham số đầu vào là giá trị nhiệt độ Fahrenheit. Hàm trả về nhiệt độ Celcius. Hiện bảng nhiệt độ Fahrenheit từ 0 đến 20 và giá trị nhiệt độ Celsius tương ứng.

Bài 9.6

- Cho 1 số $n > 0$, là 1 số có `k` chữ số. Viết hàm để kiểm tra xem `n` là số gồm toàn số chẵn hay `n` là số gồm toàn số lẻ.

Bài 9.7

- Chương trình Vietnamese Idol có 5 giám khảo, mỗi người cho điểm trong khoảng từ 0 đến 10 cho mỗi ca sĩ. Điểm cuối cùng của ca sĩ được xác định bằng cách bỏ qua điểm cao nhất và điểm thấp nhất, lấy trung bình 3 điểm còn lại. Viết chương trình dùng phương pháp này để tính điểm của các ca sĩ sử dụng các hàm sau:
 - void getJudgeData(): hỏi người dùng nhập 1 điểm của giám khảo, lưu trong 1 biến dạng tham biến. Kiểm tra điểm có hợp lệ không
 - void calcScore(): tính và hiển thị điểm trung bình của ca sĩ.

Bài 9.8. Năm nhuận

- Viết hàm *laNamNhuận* để xác định 1 năm có phải là năm nhuận hay không. Tham số truyền vào là năm. 1 năm là năm nhuận nếu:
 - Nó là bội số của 4 nhưng không là bội số của 100 hoặc
 - Nó là bội số của 400
 - Ví dụ, 1996 và 2000 là năm nhuận, nhưng 1900, 2002 và 2100 không là năm nhuận.