



BÀI THỰC HÀNH SỐ 6

1. MỤC TIÊU

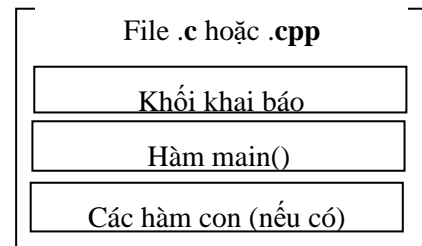
- Viết các hàm do người lập trình tự định nghĩa (**user define function**) với lệnh **while/for**.

2. LÝ THUYẾT CẦN GHI NHỚ

2.1. Cấu trúc một chương trình có sử dụng hàm trong C.

2.1.1. Cách 1: sử dụng 1 file cho toàn bộ chương trình

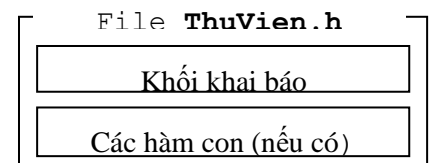
- Khởi khai báo:** Bao gồm các khai báo về:
 - Khai báo tên các thư viện chuẩn của C được sử dụng trong chương trình.
 - Hằng số sẽ sử dụng trong chương trình.
 - Các kiểu dữ liệu tự định nghĩa.
 - Các biến toàn cục.
 - Hàm con (các nguyên mẫu hàm - prototype).
- Hàm chính** (main()): Chứa các biến, các lệnh và các lời gọi hàm cần thiết trong chương trình.
- Các hàm con:**
 - Được sắp xếp sao cho mỗi hàm nằm trên 1 đoạn riêng.
 - Không đặt nội dung của hàm này chứa trong hàm khác, hoặc nội dung của 2 hàm có phần giao nhau.
 - Không cần quan tâm thứ tự sắp xếp trước/sau của các hàm.



2.1.2. Cách 2: sử dụng 2 file cho toàn bộ chương trình

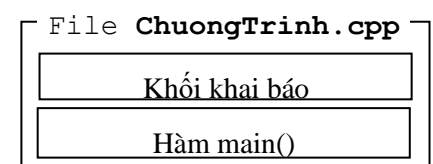
2.1.2.1. File thư viện chứa các hàm do người dùng tự tạo

- Khởi khai báo:** Bao gồm các khai báo về:
 - Khai báo tên thư viện chuẩn của ngôn ngữ C được sử dụng trong chương trình.
 - Các kiểu dữ liệu tự định nghĩa.
 - Hằng số sẽ sử dụng trong chương trình.
 - Các biến toàn cục.
 - Hàm con (các nguyên mẫu hàm - prototype).
- Các hàm con:** (tương tự như cách 1)



2.1.2.2. File chương trình

- Khởi khai báo:** khai báo tên file thư viện chứa các hàm do người dùng tự tạo. Tên file (kể cả phần mở rộng) được đặt trong dấu nháy đôi.
- Hàm chính** (main()): Chứa các biến, các lệnh và các lời gọi hàm cần thiết trong chương trình.





+ **Ước số chung lớn nhất (USCLN - Greatest Common Divisor (GCD))** của hai hay nhiều số nguyên là số nguyên dương lớn nhất là ước số chung của các số đó. Ví dụ, USCLN của ba số nguyên dương 6, 9, 15 là số 3 vì 3 là số lớn nhất mà cả ba số 6, 9, 15 cùng chia chắn.

Ví dụ tìm USCLN của 252 và 105:

các ước số của 252: **1, 2, 4, 6, 7, 9, 12, 14, 18, 21, 28, 36, 42, 63, 84, 126.**

các ước số của 105: **1, 2, 3, 5, 7, 15, 21, 35**

⇒ ước số chung của 252 và 105 là 1, 2, 7, 21 ⇒ ước số chung lớn nhất của 252 và 105 là **21**

+ **Các giải thuật tìm USCLN:**

- **Cách 1:** sử dụng phép chia lấy dư (%)

int gcd1(int a, int b)

- B1: Trong khi $a * b \neq 0$
 - B1.1: Nếu $a > b$ thì gán $a \% = b$; // $a = a \% b$
 - B1.2: ngược lại, gán $b \% = a$;
- B2: return $a + b$; //vì lúc này hoặc a hoặc b đã bằng 0.

Ví dụ tìm USCLN của 252 và 105:

Lần 1: $a=252, b=105$ do $a * b \neq 0$ và $a > b \Rightarrow a = 252 \% 105 = 42$

Lần 2: $a=42, b=105$ do $a * b \neq 0$ và $a < b \Rightarrow b = 105 \% 42 = 21$

Lần 3: $a=42, b=21$ do $a * b \neq 0$ và $a > b \Rightarrow a = 0$

Do $a * b \neq 0 \Rightarrow$ return $a + b =$ return $0 + 21 =$ return 21

- **Cách 2:** sử dụng phép trừ (-)

int gcd2(int a, int b)

- B1: Trong khi $a \neq b$
 - B1.1: nếu $a > b$ thì gán $a -= b$; // $a = a - b$
 - B1.2: Ngược lại, gán $b -= a$;
- B2: return a ; // hoặc return b , vì lúc này $a=b$

Ví dụ tìm USCLN của 252 và 105:

Lần 1: $a=252, b=105$: do $a > b \Rightarrow a = 252 - 105 = 147$; $b=105$

Lần 2: $a=147, b=105$: do $a > b \Rightarrow a = 147 - 105 = 42$; $b=105$

Lần 3: $a=42, b=105$: do $a < b \Rightarrow a = 42$; $b=105 - 42=63$

Lần 4: $a=42, b=63$: do $a < b \Rightarrow a = 42$; $b=63-42=21$

Lần 5: $a=42, b=21$: do $a > b \Rightarrow a = 42-21=21$; $b=21$.

Kết thúc do $a=b=21$

- **Cách 3: (đệ quy) Giải thuật Euclid** hay Thuật toán Euclid (Euclid là nhà toán học cổ Hy Lạp, khoảng năm 300 trước Công Nguyên): Với 2 số nguyên dương a và b ($a \geq b$) và $us=a \% b$. Khi đó, ta có:

$$USCLN(a,b) = \begin{cases} b & \text{nếu } us = 0 \\ USCLN(b, us) & \text{nếu } us \neq 0 \end{cases}$$

int gcd3(int a, int b)

- B1: Nếu $b == 0$ return a ;
- B2: return $gcd3(b, a \% b)$

Ví dụ tìm USCLN của 252 và 105:

Lần 1: $a=252, b=105 \Rightarrow us = 252 \% 105 = 42 (= 105 \times 2 + 42)$

Lần 2: $a=105, b=42 \Rightarrow us = 105 \% 42 = 21 (= 42 \times 2 + 21)$

Lần 3: $a=42, b=21 \Rightarrow us = 42 \% 21 = 0$

Lần 4: $a=21, b=0 \Rightarrow$ return $a =$ return 21



3. BÀI TẬP THỰC HÀNH CƠ BẢN

Tổ chức chương trình dưới dạng các hàm chức năng cho các bài tập sau:

Bài 1. Cho nhập số nguyên dương n , liệt kê tất cả các ước số của n .

Hướng dẫn

```
void liet_ke_uoc_so(int n){
    for(int i =1; i<=n/2; i++){
        if(n%i==0){ // la uoc so
            // in i ra màn hình
        }
    }
}
```

Bài 2. Cho nhập số nguyên dương n , đếm số lượng các ước số của n .

Hướng dẫn

- Tạo 1 biến đếm
- Sử dụng vòng for tương tự bài 1, nếu i là ước số thì tăng biến đếm lên 1

Bài 3. Cho nhập số nguyên dương n , tính tổng và tích các ước số của n .

Hướng dẫn

- Tạo 1 biến Tổng S
- Sử dụng vòng for tương tự bài 1, nếu i là ước số thì cộng i vào biến tổng S

Bài 4. Cho nhập số nguyên dương n , tính tổng các ước số chẵn của n .

Hướng dẫn

- Tạo 1 biến Tổng S
- Sử dụng vòng for tương tự bài 1, nếu i là ước số thì kiểm tra i có phải là số chẵn không, nếu phải thì cộng i vào biến tổng S

Bài 5. Cho nhập số nguyên dương n , tìm ước số lẻ lớn nhất của n (nhỏ hơn n).

Ví dụ Ước số lẻ lớn nhất của 27 là 9.

Hướng dẫn

- Khai báo biến max và gán giá trị là 1
- Sử dụng vòng for tương tự bài 1, nếu i là ước số thì kiểm tra i có phải là số lẻ không, nếu phải kiểm tra nếu $i > max$ thì gán $max = i$

Bài 6. Cho nhập 2 số nguyên dương n và m , liệt kê các ước số của n có giá trị nhỏ hơn m .

Ví dụ Nhập $n=16$, $m=7$;

in ra các ước số của 16 nhỏ hơn 7 là 1, 2, 4.



Hướng dẫn

- Sử dụng vòng for tương tự bài 1, nếu i là ước số thì kiểm tra i có nhỏ hơn m không, nếu thỏa điều kiện thì in i ra màn hình

Bài 7. Cho nhập 2 số nguyên dương a, b . Tìm USCLN của a và b .

Hướng dẫn

- Giải thuật tìm USCLN của a, b : Ước chung lớn nhất của hai số nguyên a và b là số nguyên dương lớn nhất mà a và b chia hết
- Cách làm:
 - Cách 1:
 - Trong 2 số a, b tìm số nhỏ nhất trong 2 số
 - Dùng lệnh lặp for để duyệt từ 1 đến số nhỏ nhất vừa tìm, tìm số mà cả a và b đều chia hết

```
// tìm min là số nhỏ nhất trong 2 số a,b
for(int i = 1; i <= min; i++) {
    if(a%i == 0 && b%i == 0) {
        ucln = i;
    }
}
```

- Cách 2: trừ hai số A và B cho nhau tới khi hai số này bằng nhau → tìm được ƯCLN của 2 số

```
while(a != b) {
    //Lấy số lớn trừ số bé.
    if (a > b) {
        a -= b;
    }else{
        a -= b;
    }
} // uscln là a hoặc b (vì lúc này a=b)
```

Bài 8. Hãy đếm số lượng chữ số của số nguyên dương n giống như chữ số đầu tiên của số nguyên dương n .

Ví dụ Nhập $n = 10151$. In ra màn hình: Có ba chữ số 1.



Hướng dẫn

- Viết hàm để lấy ra chữ số đầu tiên của số n

```
int lay_chuso_dau_tien (int n) {  
    while(n >= 10)  
    {  
        Thực hiện phép chia n cho 10: n =n/10  
    }  
    // lệnh while sẽ thực hiện cho đến khi n<10 → n chỉ còn 1 chữ số  
  
    return n;  
}
```

- Viết hàm để đếm trong n có bao nhiêu số giống với số vừa trả về ở hàm lay_chuso_dau_tien

```
void dem_so (int n) {  
    int first = lay_chuso_dau_tien(n);  
    int dem =0;  
    // sử dụng lệnh while để tách từng chữ số có trong n và so sánh với  
    first, nếu bằng với first thì tăng biến dem lên  
    // in ra màn hình biến dem  
}
```

Bài 9. Hãy kiểm tra các chữ của số nguyên dương n có tăng dần từ trái sang phải hay không?

Hướng dẫn

- Dùng phép mod để lấy ra chữ số cuối cùng của n
- Dùng vòng lặp while để trích xuất các số còn lại trong n và so sánh



```
while (n>=10)
{
    int donvi=n%10;
    n=n/10;
    if(donvi>n%10)
    {
        // code xử lý ở đây
    }
}
```

Bài 10. In ra cách đọc của một số dương: cho nhập số nguyên n với giá trị của n nằm trong khoảng $(-4\text{ tỷ} \leq n \leq 4\text{ tỷ})$. In ra cách đọc của n .

Ví dụ Nhập $n = -105$. In ra màn hình: Am Mot khong nam.

4. BÀI TẬP NÂNG CAO

Bài 1. Viết chương trình nhập số nguyên dương n gồm 5 chữ số, kiểm tra xem các chữ số n có phải là số đối xứng hay không.

Ví dụ Đối xứng: 13531

Không đối xứng: 13921

Bài 2. Số vừa là số may mắn, vừa là số đối xứng: Viết chương trình in ra tất cả các số từ -1 tỷ đến +1 tỷ vừa là số may mắn vừa là số đối xứng?

Ví dụ

$n=686 \Rightarrow$ là số may mắn và đối xứng.

$n=-686 \Rightarrow$ là số may mắn và đối xứng.

$n=6\textbf{8}6\textbf{6}6 \Rightarrow$ là số may mắn nhưng không phải số đối xứng.

$n=\textbf{5}66\textbf{5} \Rightarrow$ là số đối xứng nhưng không phải số may mắn.

Bài 3. Nhập vào số nguyên dương là lũy thừa của 2 (không cần viết chương trình kiểm tra; khi nhập vào, người dùng sẽ chủ động nhập 1 số sao cho số đó là lũy thừa của 2 hay nói cách khác có thể phân tích thành dạng 2^k). In ra cách biểu diễn của n dưới dạng số mũ của 2.

Ví dụ nhập $n = 8$ in ra $8 = 2^3$

Bài 4. Cho nhập số nguyên dương n ($n \geq 2$). Hãy phân tích n thành tích các thừa số nguyên tố.

Ví dụ nhập $n=1350$. In ra **$1350 = 2 * 3 * 3 * 3 * 5 * 5$**

Bài 5. Cho nhập số nguyên dương n . In ra màn hình cách phân tích n thành thừa số nguyên tố nhưng dưới dạng số mũ.



Ví dụ nhập $n=1350$. In ra $1350 = 2 * 3^3 * 5^2$

- Bài 6.** Viết chương trình cho người dùng nhập vào tử và mẫu số, thực hiện đơn giản phân số.
- Bài 7.** Cho nhập 2 số nguyên dương a, b . Tìm BSCNN của a và b .
- Bài 8.** Cho nhập 2 số nguyên dương n và m , liệt kê các bội số của n có giá trị nhỏ hơn m .

Ví dụ Nhập $n=6, m=27$;

in ra các bội số của 6 nhỏ hơn 27 là 12, 18, 24

- Bài 9.** Tìm ước chung lớn nhất và bội chung nhỏ nhất của 2 số nguyên dương n và m nhập từ bàn phím. (Sử dụng thuật toán Euclide : $USCLN(A,B) = USCLN(B, A \bmod B)$ với $A > B$)
- Bài 10.** Có 3 tuyến xe bus cùng khởi hành tại bến vào lúc 5h và chạy theo những tuyến đường (hướng đi) khác nhau. Mỗi lượt chạy, xe thứ 1 chạy và trở lại bến sau 1h05', nghỉ 10 phút rồi tiếp tục chạy lượt kế tiếp; xe thứ 2 chạy và trở lại bến sau 55', nghỉ 5 phút rồi tiếp tục chạy lượt kế tiếp; xe thứ 3 chạy và trở lại bến sau 48', nghỉ 2 phút rồi tiếp tục chạy lượt kế tiếp. Hỏi sau bao lâu nữa 3 xe sẽ lại cùng xuất phát.

Tổng quát, cho người dùng nhập thời gian chạy 1 lượt, thời gian nghỉ của mỗi loại xe. Cho biết sau bao lâu 3 xe sẽ lại cùng xuất phát.