

## DANH SÁCH BÀI TẬP – BUỔI 2- CÁC PHÉP TOÁN XỬ LÝ TRÊN BIT

**Bài 1: Viết chương trình chuyển từ số thập phân n sang số nhị phân ( $0 \leq n \leq 255$ ).**

- Sử dụng phương pháp che bit.

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

```
int main(){
    int n;

    printf ("Nhap gia tri can chuyen doi:");scanf("%d",&n);

    if(n>=0 && n<=255){
        printf("So nhi phan sau khi chuyen doi la:");
        if(n & 0x80)printf("1"); else printf("0");
        if(n & 0x40)printf("1"); else printf("0");
        if(n & 0x20)printf("1"); else printf("0");
        if(n & 0x10)printf("1"); else printf("0");
        if(n & 0x08)printf("1"); else printf("0");
        if(n & 0x04)printf("1"); else printf("0");
        if(n & 0x02)printf("1"); else printf("0");
        if(n & 0x01)printf("1"); else printf("0");
    }else{
        printf("So khong hop le");
    }

    return 0;
}
```

**Bài 2: Viết chương trình nhập vào 1 số n từ bàn phím ( $0 \leq n \leq 255$ ). Sử dụng phép dịch bit, in ra kết quả.**

**\* code 1:**

```
#include <stdio.h>

int main(){

    int a=20;

    printf("Ket qua: %d",a<<3);

    return 0;

}
```

**\* code 2:**

```
#include <stdio.h>

int main(){

    int a=10;

    printf("Ket qua: %d",a>>2<<1+1);

    return 0;

}
```

**Bài 3: Viết chương trình nhập vào 2 số a,b từ bàn phím. Lần lượt thực hiện phép toán AND, OR, XOR, NOT trên 2 số đó, in ra kết quả.**

```
#include <stdio.h>

int main(){

    int a=10;

    int b=20;

    printf("a & b = %d",a&b);

    return 0;

}
```

**Bài 4: In ra bảng giá trị chân lý của các phép toán AND, OR, XOR, NOT.**

**Bài 5: Nhập vào 2 số a,b từ bàn phím. Tìm số lớn nhất trong 2 số đó, in kết quả ra màn hình.**

**Bài 6: Nhập vào 3 số a,b,c từ bàn phím. Tìm số lớn nhất trong 3 số đó, in kết quả ra màn hình.**

**Bài 7: Nhập vào 4 số a,b,c,d từ bàn phím. Tìm số lớn nhất trong 4 số đó, in kết quả ra màn hình.**

**Bài 8: Nhập vào 2 hệ số a,b từ bàn phím. Thực hiện giải phương trình bậc nhất  $ax+b=0$ . In kết quả ra màn hình.**

**Bài 9: Nhập vào 3 hệ số a,b,c từ bàn phím. Thực hiện giải phương trình bậc 2:  $ax^2+bx+c=0$ . In kết quả ra màn hình.**

**Bài 10. Viết chương trình nhập vào một số thực a. In ra màn hình kết quả nghịch đảo của a khi  $a \neq 0$ , khi  $a = 0$  in ra thông báo “Khong the tim duoc nghich dao cua a”**

```
#include <stdio.h>

int main(){

float a;

printf ("Nhap a="); scanf ("%f",&a);

if (a != 0)

printf ("Nghich dao cua %2.2f la %2.2f",a,1/a);

else

printf ("Khong the tim duoc nghich dao cua a");

return 0;

}
```

**Bài 11: Viết chương trình nhập vào hai số ở hệ hexa rồi thực hiện các phép toán xử lý bit trên hai số đó.**

```
#include "stdio.h"

#define OR 1
```

```

#define AND 2

#define EX_OR 3

#define NOR 4

#define NAND 5

int main (){

int value1, value2, option;

printf ("Nhap vao hai so he hexa:");

scanf ("%x %x",&value1, &value2);

printf ("Nhap vao phep toan van thao tac");

printf ("1-OR\n2-AND\n3-EXOR\n4-NOR\n5-NAND");

scanf ("%d",&option);

if (option == OR)

printf ("Ket qua: %x (hex):",value1 | value2);

else if (option == AND)

printf ("Ket qua: %x (hex):",value1 & value2);

else if (option == EX_OR)

printf ("Ket qua: %x (hex):",value1 ^ value2);

else if (option == NOR)

printf ("Ket qua: %x (hex):",~(value1 | value2));

else if (option == NAND)

printf ("Ket qua: %x (hex)",~(value1 & value2));

return (0); }

```

### **Bài 12: Viết chương trình tính tiền cước TAXI. Biết rằng:**

- Km đầu tiên là 5000đ
- 30Km tiếp theo là 4000đ

- Nếu lớn hơn 30Km thì mỗi Km thêm ra sẽ phải trả là 3000đ
- Hãy nhập số Km sau đó in ra số tiền phải trả.

```
#include <stdio.h>

int main(){

    float TotalKm;

    float Cost = 0;

    printf("\n Nhập vào số Km: ");

    scanf("%f", &TotalKm);

    if(TotalKm <= 1){

        Cost = TotalKm * 5000; }

    else if(TotalKm <= 30 && TotalKm > 1) {

        Cost = (TotalKm - 1)*4000 + 1*5000;

    } else {

        Cost = 1*5000 + 29*4000 + (TotalKm - 30)*3000;    }

    printf("\n Số tiền phải trả: %0.3f VND", Cost);

    return 0;

}
```

### **Bài 13: Kiểm tra sự tồn tại của 1 tam giác**

- Nhập 3 số nguyên dương
- Điều kiện để 1 tam giác tồn tại là: tổng 2 cạnh bất kì luôn lớn hơn cạnh còn lại.

```
#include <stdio.h>

int main(){

    float a, b, c;

    do{
```

```

printf("\nNhap a: ");

scanf("%f", &a);

printf("\nNhap b: ");

scanf("%f", &b);

printf("\nNhap c: ");

scanf("%f", &c);

}while (a <= 0 || b <= 0 || c <= 0);

if((a + b > c) && (a + c > b) && (b + c > a))

    printf("\nDay la tam giac");

else

    printf("\nDay khong phai la tam giac");

return 0;

}

```

#### **Bài 14: Xác định màu của các vòng màu trên điện trở.**

```

#include "stdio.h"

int main (){

unsigned int colour;

printf ("Nhap vao so cac vong mau (0->9):");

scanf("%u",&colour);

printf("Vong mau la mau:");

switch (colour){

case 0: printf("BLACK"); break;

case 1: printf("BROWN"); break;

case 2: printf("RED"); break;

```

```

case 3: printf("OREANGE"); break;
case 4: printf("YELLOW"); break;
case 5: printf("GREEN"); break;
case 6: printf("BLUE"); break;
case 7: printf("VIOLET"); break;
case 8: printf("GREY"); break;
case 9: printf("WHITE"); break;
}
return 0;
}

```

**Bài 15: Tạo ra 1 menu cho phép lựa chọn, nếu chọn số 1 thì thực hiện giải phương trình bậc 1; nếu chọn số 2 thì thực hiện giải phương trình bậc 2.**

**Bài 16: Thực hiện tính các biểu thức sau:**

**Tính  $S(n) = 1 + 2 + 3 + \dots + n$**

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

```
int main(){
```

```
    int i, n;
```

```
    long S;
```

```
    S = 0;
```

```
    i = 1;
```

```
    printf("\nNhap n: ");
```

```
    scanf("%d", &n);
```

```
    while(i <= n){
```

```

        S = S + i;

        i++;

    }

    printf("\nTong 1 + 2 + ... + %d la %ld", n, S);

    return 0;

}

```

- Tính  $S(n) = 1^2 + 2^2 + \dots + n^2$

- Tính  $S(n) = 1 + \frac{1}{2} + \frac{1}{3} + \dots + \frac{1}{n}$

- Tính  $S(n) = \frac{1}{2} + \frac{1}{4} + \dots + \frac{1}{2n}$

- Tính  $S(n) = 1 + \frac{1}{3} + \frac{1}{5} + \dots + \frac{1}{(2n+1)}$

**Bài 17:** Nhập vào 1 số nguyên n từ bàn phím; thực hiện tính tổng các phần tử từ 1 đến n, tổng các phần tử chẵn từ 1 đến n, tổng các phần tử lẻ, tổng các phần tử chia hết cho 3, tổng các phần tử chia hết cho 5.

**Bài 18:** Nhập vào 1 số n từ bàn phím. Đếm số nguyên tố từ 1 đến n.

**Bài 19:** Nhập vào 1 số n từ bàn phím. Đếm số hoàn hảo từ 1 đến n.

**Bài 20:** Nhập vào 1 số n từ bàn phím. Đếm số chính phương từ 1 đến n.