

1. Write a C program using function that takes a character indicating a color as an input argument and displays the full color if it is present else prints the message that the color is not present. Here function doesn't return anything. Rainbow colours: R O Y G B I V = Red Orange Yellow Green Blue Indigo Violet.

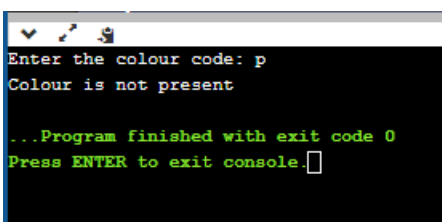
Program:

```
#include <stdio.h>

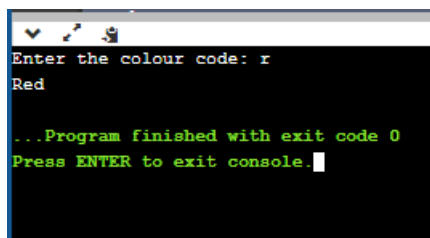
void colour(char c)
{
    switch(c)
    {
        case 'r':
        case 'R':
            printf("Red");
            break;
        case 'o':
        case 'O':
            printf("Orange");
            break;
        case 'y':
        case 'Y':
            printf("Yellow");
            break;
        case 'g':
        case 'G':
            printf("Green");
            break;
        case 'b':
        case 'B':
```

```
    printf("Blue");  
    break;  
    case 'i':  
    case 'I':  
        printf("Indigo");  
        break;  
    case 'v':  
    case 'V':  
        printf("Violet");  
        break;  
    default:  
        printf("Colour is not present");  
    }  
}  
  
int main()  
{  
    char c;  
    printf("Enter the colour code: ");  
    scanf("%c",&c);  
    colour(c);  
    return 0;  
}
```

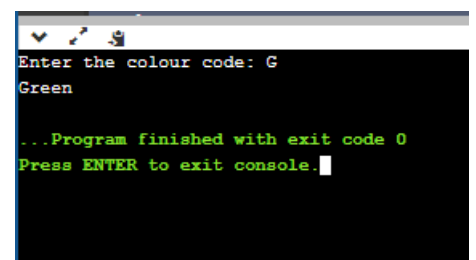
OUTPUT:



```
Enter the colour code: p  
Colour is not present  
  
...Program finished with exit code 0  
Press ENTER to exit console.
```



```
Enter the colour code: r  
Red  
  
...Program finished with exit code 0  
Press ENTER to exit console.
```



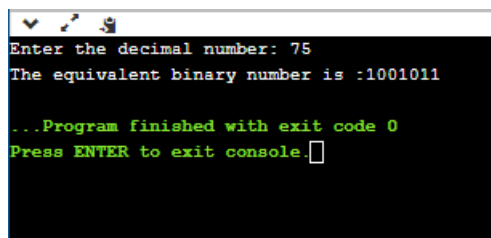
```
Enter the colour code: G  
Green  
  
...Program finished with exit code 0  
Press ENTER to exit console.
```

2. Write a program in C to convert decimal number to binary number using the function.Program:

```
#include <stdio.h>

void bin(int n)
{
    int arr[32];
    int i = 0,j;
    while (n > 0)
    {
        arr[i] = n % 2;
        n = n / 2;
        i++;
    }
    for (j = i - 1; j >= 0; j--)
        printf("%d",arr[j]);
}

int main()
{
    int n;
    printf("Enter the decimal number: ");
    scanf("%d",&n);
    printf("The equivalent binary number is :");
    bin(n);
    return 0;
}
```

OUTPUT:

```
Enter the decimal number: 75
The equivalent binary number is :1001011

...Program finished with exit code 0
Press ENTER to exit console.
```

3. Write a function to find if a given number is prime or not. Function takes number as input argument. It returns 1 if number is prime else return 0. Based on the return value, print the appropriate message in main function.

Program:

```
#include <stdio.h>

int prime(int n)
{
    int i,t=0;
    for(i=2;i<=n/2;i++)
    {
        if(n%i==0)
        {
            t=1;
            break;
        }
    }
    if(t==1)
        return 0;
    else
        return 1;
}

int main()
{
```

```
int n;

printf("Enter the number:");

scanf("%d",&n);

if(n==1)

    printf("The number is neither prime nor composite");

else if(prime(n))

    printf("The number is prime");

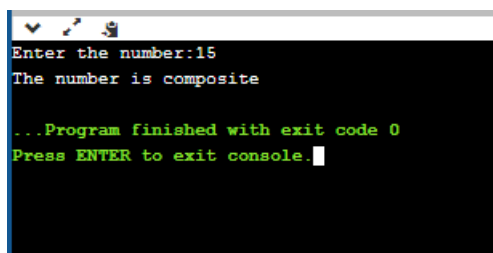
else

    printf("The number is composite");

return 0;

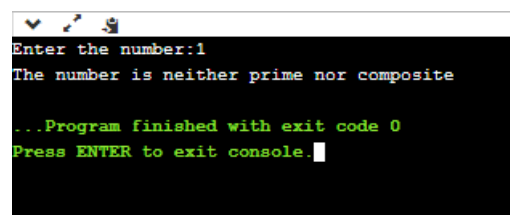
}
```

OUTPUT:



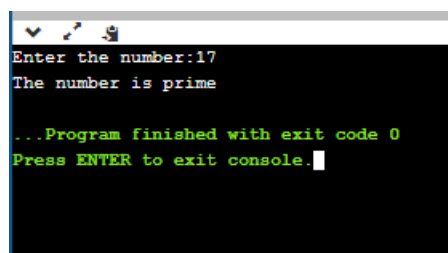
```
Enter the number:15
The number is composite

...Program finished with exit code 0
Press ENTER to exit console.
```



```
Enter the number:1
The number is neither prime nor composite

...Program finished with exit code 0
Press ENTER to exit console.
```



```
Enter the number:17
The number is prime

...Program finished with exit code 0
Press ENTER to exit console.
```

4. Write a program that takes the x-y coordinates of a point in the Cartesian plane. Write a function which takes as input the x-y coordinates and returns the quadrant in which it lies.

Program:

```
#include <stdio.h>

int quad(int x,int y)

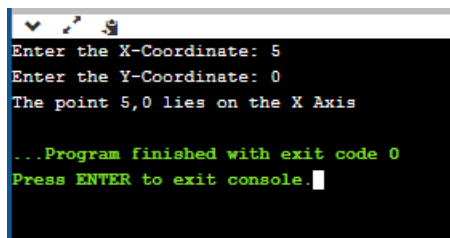
{

    if(x>0)
```

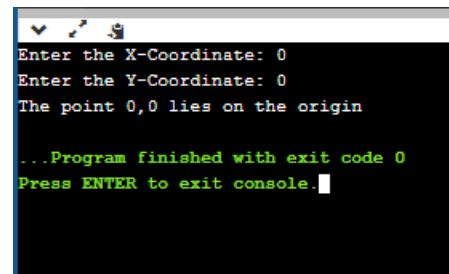
```
{
    if(y>0)
        return 1;
    else
        return 4;
}
else
{
    if(y>0)
        return 2;
    else
        return 3;
}
}
int main()
{
    int x,y;
    printf("Enter the X-Coordinate: ");
    scanf("%d",&x);
    printf("Enter the Y-Coordinate: ");
    scanf("%d",&y);
    if(x==0 && y==0)
        printf("The point %d,%d lies on the origin",x,y);
    else if(x==0)
        printf("The point %d,%d lies on the Y Axis",x,y);
    else if(y==0)
        printf("The point %d,%d lies on the X Axis",x,y);
```

```
else  
    printf("The quadrant in which point %d,%d lies is %d",x,y,quad(x,y));  
return 0;  
}
```

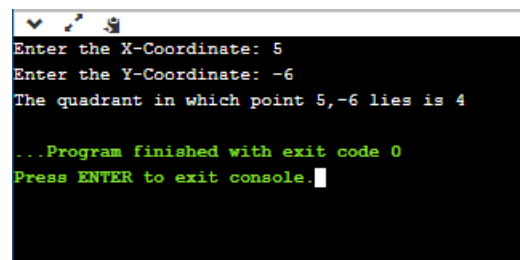
OUTPUT:



```
Enter the X-Coordinate: 5  
Enter the Y-Coordinate: 0  
The point 5,0 lies on the X Axis  
  
...Program finished with exit code 0  
Press ENTER to exit console.
```



```
Enter the X-Coordinate: 0  
Enter the Y-Coordinate: 0  
The point 0,0 lies on the origin  
  
...Program finished with exit code 0  
Press ENTER to exit console.
```



```
Enter the X-Coordinate: 5  
Enter the Y-Coordinate: -6  
The quadrant in which point 5,-6 lies is 4  
  
...Program finished with exit code 0  
Press ENTER to exit console.
```