if-else

1. Write a C program to input temperature in Centigrade and convert to Fahrenheit.

Iutput:

```
#include <stdio.h>
int main() {
     float c, f;
     printf("Enter temperature in Centigrade: ");
     scanf("%f", &c);
     f = (c * 9 / 5) + 32;
     printf("Temperature in Fahrenheit = %.2f", f);
     return 0;
}
```

Output:

```
Enter temperature in Centigrade: 100
Temperature in Fahrenheit = 212.00
Process returned 0 (0x0)   execution time : 23.013 s
Press any key to continue.
```

2. Write a C program to input radius of a circle from user and find diameter, circumference and area of circle.

Iutput:

```
#include <stdio.h>
#define PI 3.1416
int main() {
     float r, d, c, a;
     printf("Enter radius of circle: ");
     scanf("%f", &r);
     d = 2 * r;
     c = 2 * PI * r;
     a = PI * r * r;
     printf("Diameter = %.2f\nCircumference = %.2f\nArea =
     %.2f", d, c, a);
     return 0;
}
```

Output:

```
Enter radius of circle: 10
Diameter = 20.00
Circumference = 62.83
Area = 314.16
Process returned 0 (0x0)   execution time : 25.181 s
Press any key to continue.
```

3. Write a C program to input any two numbers from user and swap values of both numbers using third variable, bitwise operator and arithmetic operators.

Iutput:

```
#include <stdio.h>
int main() {
     int a, b, temp;
     printf("Enter two numbers: ");
     scanf("%d %d", &a, &b);

     // Using third variable
     temp = a;
     a = b;
     b = temp;
     printf("After swapping: a = %d, b = %d\n", a, b);
     return 0;
}
```

Output:

```
Enter two numbers: 5 7
After swapping: a = 7, b = 5

Process returned 0 (0x0)   execution time : 26.846 s
Press any key to continue.
```

Iutput:

```
#include <stdio.h>
int main() {
     int a, b;
     printf("Enter two numbers: ");
     scanf("%d %d", &a, &b);

     // Bitwise XOR method
     a = a ^ b;
     b = a ^ b;
     a = a ^ b;
     printf("After swapping: a = %d, b = %d\n", a, b);
```

```
        return 0;
    }
```

Output:

```
Enter two numbers: 5 7
After swapping: a = 7, b = 5

Process returned 0 (0x0)   execution time : 26.278 s
Press any key to continue.
```

Iutput:

```
#include <stdio.h>
int main() {
    int a, b;
    printf("Enter two numbers: ");
    scanf("%d %d", &a, &b);

    // Using arithmetic operators
    a = a + b;
    b = a - b;
    a = a - b;

    printf("After swapping: a = %d, b = %d\n", a, b);
    return 0;
}
```

Output:

```
Enter two numbers: 5 7
After swapping: a = 7, b = 5

Process returned 0 (0x0)   execution time : 15.706 s
Press any key to continue.
```

Iutput:

```
#include <stdio.h>
int main() {
    int a, b;
    printf("Enter two numbers: ");
    scanf("%d %d", &a, &b);

    // Using arithmetic operators
    a = a - b;
    b = a + b;
    a = b - a;

    printf("After swapping: a = %d, b = %d\n", a, b);
```

```
        return 0;
    }
```

Output:

```
Enter two numbers: 5 7
After swapping: a = 7, b = 5

Process returned 0 (0x0)   execution time : 13.512 s
Press any key to continue.
```

Iutput:

```
#include <stdio.h>
int main() {
    int a, b;
    printf("Enter two numbers: ");
    scanf("%d %d", &a, &b);

    // Using arithmetic operators
    a = a * b;
    b = a / b;
    a = a / b;

    printf("After swapping: a = %d, b = %d\n", a, b);
    return 0;
}
```

[Works only if: a and b are **non-zero integers** and the multiplication doesn't overflow]

Output:

```
Enter two numbers: 5 7
After swapping: a = 7, b = 5

Process returned 0 (0x0)   execution time : 10.009 s
Press any key to continue.
```

Iutput:

```
#include <stdio.h>
int main() {
    int a, b;
    printf("Enter two numbers: ");
    scanf("%d %d", &a, &b);

    // Using arithmetic operators
    a = a / b;
    b = a * b;
    a = b / a;

    printf("After swapping: a = %d, b = %d\n", a, b);
```

```
        return 0;
}
```

[Works only if: a and b are **non-zero integers** and the multiplication doesn't overflow]

Output:

```
Enter two numbers: 9 8
After swapping: a = 8, b = 8

Process returned 0 (0x0)   execution time : 15.275 s
Press any key to continue.
```

4. Write a C program to find maximum between three numbers using ladder if else or nested if.

Iutput:

```
#include <stdio.h>
int main() {
    int a, b, c;
    printf("Enter three numbers: ");
    scanf("%d %d %d", &a, &b, &c);

    if(a >= b && a >= c)
        printf("Maximum = %d", a);
    else if(b >= a && b >= c)
        printf("Maximum = %d", b);
    else
        printf("Maximum = %d", c);

    return 0;
}
```

Output:

```
Enter three numbers: 8 3 5
Maximum = 8
Process returned 0 (0x0)   execution time : 15.205 s
Press any key to continue.
```

Output:

```
Enter three numbers: 5 8 3
Maximum = 8
Process returned 0 (0x0)   execution time : 19.292 s
Press any key to continue.
```

Output:

```
Enter three numbers: 3 5 8
Maximum = 8
Process returned 0 (0x0)   execution time : 57.390 s
Press any key to continue.
```

5. Write a C program to check a given year is leap year or not.

Iutput:

```
#include <stdio.h>
int main() {
     int year;
     printf("Enter a year: ");
     scanf("%d", &year);

     if((year % 400 == 0) || (year % 4 == 0 && year % 100
     != 0))
          printf("%d is a Leap Year", year);
     else
          printf("%d is not a Leap Year", year);

     return 0;
}
```

Output:

```
Enter a year: 2000
2000 is a Leap Year
Process returned 0 (0x0)   execution time : 15.001 s
Press any key to continue.
```

Output:

```
Enter a year: 1927
1927 is not a Leap Year
Process returned 0 (0x0)   execution time : 26.962 s
Press any key to continue.
```

6. Write a C program to input a character from user and check whether given character is alphabet, uppercase, lowercase, digit or special character.

Iutput:

```
#include <stdio.h>
int main() {
     char ch;
     printf("Enter a character: ");
     scanf("%c", &ch);
```

```
        if(ch >= 'A' && ch <= 'Z')
            printf("Uppercase Alphabet");
        else if(ch >= 'a' && ch <= 'z')
            printf("Lowercase Alphabet");
        else if(ch >= '0' && ch <= '9')
            printf("Digit");
        else
            printf("Special Character");

        return 0;
    }
```

Output:

```
Enter a character: H
Uppercase Alphabet
Process returned 0 (0x0)   execution time : 16.017 s
Press any key to continue.
```

Output:

```
Enter a character: h
Lowercase Alphabet
Process returned 0 (0x0)   execution time : 11.406 s
Press any key to continue.
```

Output:

```
Enter a character: 79
Digit
Process returned 0 (0x0)   execution time : 10.691 s
Press any key to continue.
```

Output:

```
Enter a character: ]
Special Character
Process returned 0 (0x0)   execution time : 11.645 s
Press any key to continue.
```

7. Write a C program to check whether an alphabet is vowel or consonant.

Iutput:

```
#include <stdio.h>
int main() {
    char ch;
    printf("Enter an alphabet: ");
    scanf(" %c", &ch);
```

```
        if(ch=='a'||ch=='e'||ch=='i'||ch=='o'||ch=='u'||
        ch=='A'||ch=='E'||ch=='I'||ch=='O'||ch=='U')
             printf("%c is a Vowel", ch);
        else
             printf("%c is a Consonant", ch);

        return 0;
   }
```

Output:

```
Enter an alphabet: u
u is a Vowel
Process returned 0 (0x0)   execution time : 36.355 s
Press any key to continue.
```

Output:

```
Enter an alphabet: J
J is a Consonant
Process returned 0 (0x0)   execution time : 23.990 s
Press any key to continue.
```

---

loop

1. Write a C program to print natural numbers in range N.

Iutput:

```
#include <stdio.h>
int main() {
     int i, n;
     printf("Enter the value of N: ");
     scanf("%d", &n);

     for(i=1; i<=n; i++) {
          printf("%d ", i);
     }
     return 0;
}
```

Output:

```
Enter the value of N: 5
1 2 3 4 5
Process returned 0 (0x0)   execution time : 35.414 s
Press any key to continue.
```

2. Write a C program to print all alphabets in uppercase and lowercase using while loop.

Iutput:

```
#include <stdio.h>
int main() {
     char ch = 'A';

     printf("Uppercase Alphabets:\n");
     while(ch <= 'Z') {
          printf("%c ", ch);
          ch++;
     }

     ch = 'a';
     printf("\nLowercase Alphabets:\n");
     while(ch <= 'z') {
          printf("%c ", ch);
          ch++;
     }

     return 0;
}
```

Output:

```
Uppercase Alphabets:
A B C D E F G H I J K L M N O P Q R S T U V W X Y Z
Lowercase Alphabets:
a b c d e f g h i j k l m n o p q r s t u v w x y z
Process returned 0 (0x0)   execution time : 1.945 s
Press any key to continue.
```

3. Write a C program to print all even and odd numbers from 1 to n. Also calculate their sum.

Iutput:

```
#include <stdio.h>
int main() {
     int i, n, evenSum=0, oddSum=0;
     printf("Enter the value of n: ");
     scanf("%d", &n);

     printf("Even numbers: ");
     for(i=1; i<=n; i++) {
          if(i%2==0) {
```

```
                printf("%d ", i);
                evenSum += i;
            }
        }

        printf("\nOdd numbers: ");
        for(i=1; i<=n; i++) {
            if(i%2!=0) {
                printf("%d ", i);
                oddSum += i;
            }
        }

        printf("\nSum of even numbers = %d", evenSum);
        printf("\nSum of odd numbers = %d", oddSum);

        return 0;
    }
```

Output:

```
Enter the value of n: 6
Even numbers: 2 4 6
Odd numbers: 1 3 5
Sum of even numbers = 12
Sum of odd numbers = 9
Process returned 0 (0x0)   execution time : 17.359 s
Press any key to continue.
```

4. Write a C program to input a number and calculate its factorial.

Iutput:

```
#include <stdio.h>
int main() {
    int n, i, fact=1;
    printf("Enter a number: ");
    scanf("%d", &n);

    for(i=1; i<=n; i++) {
        fact *= i;
    }
    printf("Factorial of %d = %d", n, fact);

    return 0;
}
```

Output:

```
Enter a number: 4
Factorial of 4 = 24
Process returned 0 (0x0)   execution time : 12.845 s
Press any key to continue.
```

Or, Write a C program to multiply all natural numbers.

Iutput:

```
#include <stdio.h>
int main() {
     int n, i, product=1;
     printf("Enter a number: ");
     scanf("%d", &n);

     for(i=1; i<=n; i++) {
          product *= i;
     }
     printf("Multiplication result = %d", product);

     return 0;
}
```

Output:

```
Enter a number: 4
Multiplication result = 24
Process returned 0 (0x0)   execution time : 8.908 s
Press any key to continue.
```

5. Print the following patterns:

(i)

```
*
**
***
****
*****
```

Iutput:

```
#include <stdio.h>
int main() {
    int i, j;
    for(i=1; i<=5; i++) {
        for(j=1; j<=i; j++) {
            printf("*");
        }
        printf("\n");
```

```
        }
        return 0;
    }
```

Output:

```
*
**
***
****
*****

Process returned 0 (0x0)   execution time : 1.600 s
Press any key to continue.
```

(ii)

12345
1234
123
12
1

Iutput:

```
#include <stdio.h>
int main() {
    int i, j;
    for(i=5; i>=1; i--) {
        for(j=1; j<=i; j++) {
            printf("%d", j);
        }
        printf("\n");
    }
    return 0;
}
```

Output:

```
12345
1234
123
12
1

Process returned 0 (0x0)   execution time : 1.503 s
Press any key to continue.
```