

Aim:

Write a program to read two integer values and an arithmetic operator, depending on the operator perform different arithmetic operations.

If integer values **2** and **3** are given with operator **+**, then the output should be $2 + 3 = 5$.

If integer values **6** and **3** are given with operator **/**, then the output should be $6 / 3 = 2$.

If other than arithmetic operator is given, then display **"Error! Operator is not correct"**.

Note: Space before %c removes any white space (blanks, tabs, or newlines). It means %c without space will read white space like new line(\n), spaces(' ') or tabs(\t). By adding space before %c, we are skipping this and reading only the char given.

Instruction: To run your custom test cases strictly map your input and output layout with the visible test cases.

Source Code:

Program406.c

```
#include<stdio.h>
int main()
{
    int a,b;
    char c;
    printf("Values: ");
    scanf("%d%d",&a,&b);
    printf("Operator: ");
    scanf("\n%c",&c);
    switch(c)
    {
        case '+': printf("%d + %d = %d\n",a,b,a+b);
                break;
        case '-': printf("%d - %d = %d\n",a,b,a-b);
                break;
        case '*': printf("%d * %d = %d\n",a,b,a*b);
                break;
        case '/': if(b==0)
                printf("Division is not possible! Divide by zero error");
                else
                printf("%d / %d = %d\n",a,b,a/b);
                break;
        case '%': if(b==0)
                printf("Modulo division is not possible! Divide by zero error");
                else
                printf("%d %% %d = %d\n",a,b,a%b);
                break;
        default:printf("Invalid Operator");
                break;
    }
}
```

Execution Results - All test cases have succeeded!

Test Case - 1
User Output
Values: 6 9
Operator: -
6 - 9 = -3

Test Case - 2
User Output
Values: 6 9
Operator: *
6 * 9 = 54

Test Case - 3
User Output
Values: 8 9
Operator: @
Invalid Operator

Test Case - 4
User Output
Values: 12 0
Operator: /
Division is not possible! Divide by zero error

Test Case - 5
User Output
Values: 5 0
Operator: %
Modulo division is not possible! Divide by zero error