

COMPUTER PROGRAMING

Name: Kakavakam Jaswanth Sai

Roll no: CH.EN.U4CSE20130

Section: CSE-B

Subject code: 19CSE102 LAB

A. SIMPLE INTEREST.

Aim:

To write a C program for calculating Simple interest.

Algorithm:

Step 1: Start

Step 2: Declare Three Floating values P (Principal Amount), N (Time Period of the Loan/Deposit in years), R (Rate of interest) & SI (Simple interest).

Step 3: Input the values of P, N & R.

Step 4: Calculate $(P \times N \times R) / 100$

Step 5: Store the output of step 4 to "SI".

Step 6: Print SI.

Step 7: Stop

Program:

```
#include<stdio.h>
#include<conio.h>
void main()
{
    float P,N,R;
    float SI;
    printf("\n\n----- SIMPLE INTEREST CALCULATION ----- \n\n");
    printf("Enter the values of Principal Amount : \n");
    scanf("%f",&P);
    printf("Enter the values of Time Period of the Loan/Deposit in years : \n");
    scanf("%f",&N);
    printf("Enter the values of Rate of interest : \n");
    scanf("%f",&R);
    SI = (P * N * R) / 100;
    printf("simple interest = %.4f",SI);
}
```

Output:

```
----- SIMPLE INTEREST CALCULATION -----
Enter the values of Principal Amount :
260
Enter the values of Time Period of the Loan/Deposit in years :
6
Enter the values of Rate of interest :
2.6
simple interest = 40.5600
Process returned 25 (0x19)    execution time : 1500.525 s
Press any key to continue.
```

Result:

Thus the program to calculate the simple interest in C language has been executed and verified successfully.

B. GCD OF TWO NUMBERS

Aim:

To write a C program for calculating GCD of two Numbers.

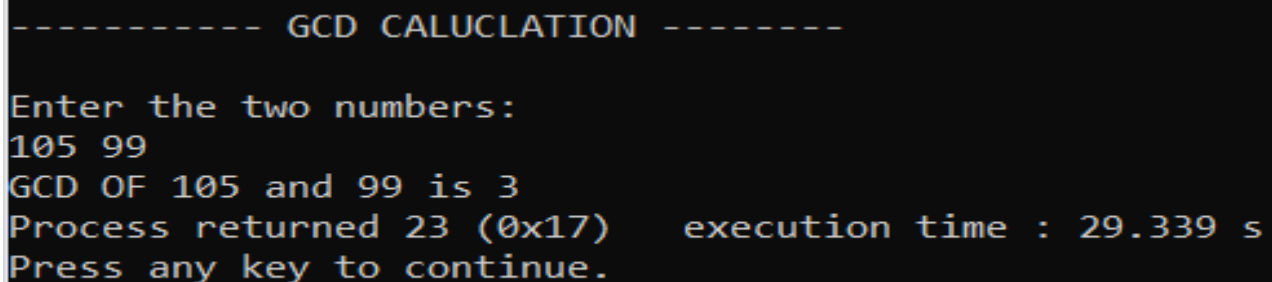
Algorithm:

- Step 1: Start
- Step 2: Declare two integers a & b.
- Step 3: Get the values of a & b.
- Step 4: While a is not equal to b
- Step 5: if (a>b) Calculate a-b and store to a
- Step 6: else Calculate b-a and store to b
- Step 7: END While
- Step 8: Print b.
- Step 9: Stop

Program:

```
#include<stdio.h>
void main()
{
    int num_1,num_2,a,b;
    printf("\n\n----- GCD |CALUCLATION ----- \n\n");
    printf("Enter the two numbers:\n");
    scanf("%d\n%d",&num_1,&num_2);
    a = num_1;
    b = num_2;
    num_1 = ( num_1 > 0 ) ? num_1 : -num_1;
    num_2 = ( num_2 > 0 ) ? num_2 : -num_2;
    while(num_1!=num_2)
    {
        if (num_1>num_2)
            num_1 = num_1 - num_2;
        else
            num_2 = num_2 - num_1;
    }
    printf("GCD OF %d and %d is %d ",a,b,num_1);
}
```

Output:



```
----- GCD CALUCLATION -----
Enter the two numbers:
105 99
GCD OF 105 and 99 is 3
Process returned 23 (0x17)    execution time : 29.339 s
Press any key to continue.
```

Result:

Thus the program to calculate the GCD of two Numbers in C language has been executed and verified successfully.

C. Demonstration of ARITHMETIC OPERATORS.

Aim:

To write a C program for demonstration of Arithmetic operators.

Algorithm:

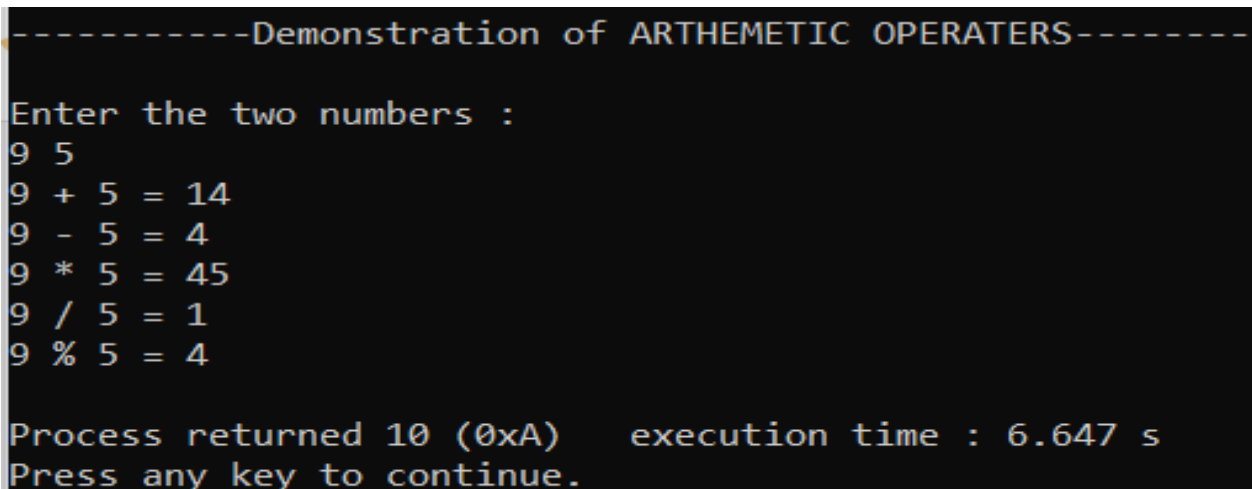
- Step 1: Start
- Step 2: Declare two integers a & b.
- Step 3: Get the values of a & b.
- Step 4: Add a and b
- Step 5: Subtract a and b
- Step 6: Multiply a and b
- Step 7: Divide a and b
- Step 8: Calculate Mod of a and b
- Step 9: Print the values of step 4,5,6,7,8.
- Step 10: Stop

Program:

```
#include<stdio.h>

void main()
{
    int a ,b;
    printf("\n\n-----Demonstration of ARTHMETIC OPERATERS-----\n\n");
    printf("Enter the two numbers :\n");
    scanf("%d\n%d",&a,&b);
    printf("%d + %d = %d\n",a,b,a+b);
    printf("%d - %d = %d\n",a,b,a-b);
    printf("%d * %d = %d\n",a,b,a*b);
    printf("%d / %d = %d\n",a,b,a/b);
    printf("%d %% %d = %d\n",a,b,a%b);
}
```

Output:



```
-----Demonstration of ARTHMETIC OPERATERS-----
Enter the two numbers :
9 5
9 + 5 = 14
9 - 5 = 4
9 * 5 = 45
9 / 5 = 1
9 % 5 = 4

Process returned 10 (0xA)   execution time : 6.647 s
Press any key to continue.
```

Result:

Thus the program to demonstration of Arithmetic operators in C language has been executed and verified successfully.

D. Demonstration of LOGICAL OPERATORS.

Aim:

To write a C program for demonstration of Logical operators.

Algorithm:

- Step 1: Start
- Step 2: Declare two integers a & b.
- Step 3: Get the values of a & b.
- Step 4: calculate the truth value of: a && b.
- Step 5: calculate the truth value of: a || b.
- Step 6: calculate the truth value of: !a or !b.
- Step 7: Print the values of step 4,5,6.
- Step 8: Stop

Program:

```
#include<stdio.h>
void main()
{
    int a,b;
    printf("\n\n-----Demonstration of LOGICAL OPERATORS-----\n\n");
    printf("Enter the two numbers: \n");
    scanf("%d\n%d",&a,&b);
    printf("%d && %d = %d\n",a,b,a&&b);
    printf("%d || %d = %d\n",a,b,a||b);
    printf("!%d = %d\n",a,!a);
}
```

Output:

```
-----Demonstration of LOGICAL OPERATORS-----
Enter the two numbers:
6 -2
6 && -2 = 1
6 || -2 = 1
!6 = 0

Process returned 7 (0x7)   execution time : 13.116 s
Press any key to continue.
```

Result:

Thus the program to demonstration of Logical operators in C language has been executed and verified successfully.

E. Demonstration of INCREMENT& DECREMENT OPERATORS.

Aim:

To write a C program for demonstration of Increment & Decrement operators.

Algorithm:

- Step 1: Start
- Step 2: Declare two integers a & b.
- Step 3: Get the values of a & b.
- Step 4: Pre increment the a and print the value.
- Step 5: Post increment the b and print the value.
- Step 6: Print the new value of a and b.
- Step 7: Pre decrement the a and print the value.
- Step 8: Post decrement the b and print the value.
- Step 9: Stop.

Program:

```
#include<stdio.h>
void main()
{
    int a,b;
    printf("\n\n-----Demonstration of INCREMENT&DECREMENT OPERATORS-----\n\n");
    printf("Enter the two numbers: \n");
    scanf("%d\n%d",&a,&b);
    printf("the value of a = %d\n",a);
    printf("the value of ++a = %d\n",++a);
    printf("New value of a = %d\n",a);
    printf("the value of b = %d\n",b);
    printf("the value of b++ = %d\n",b++);
    printf("New value of b = %d\n",b);
    printf("the value of a = %d\n",a);
    printf("the value of --a = %d\n",--a);
    printf("New value of a = %d\n",a);
    printf("the value of b = %d\n",b);
    printf("the value of b-- = %d\n",b--);
    printf("New value of b = %d\n",b);
}
```

Output:

```
-----Demonstration of INCREMENT&DECREMENT OPERATORS-----
Enter the two numbers:
2 -4
the value of a = 2
the value of ++a = 3
New value of a = 3
the value of b = -4
the value of b++ = -4
New value of b = -3
the value of a = 3
the value of --a = 2
New value of a = 2
the value of b = -3
the value of b-- = -3
New value of b = -4

Process returned 20 (0x14)    execution time : 9.396 s
Press any key to continue.
```

Result:

Thus the program to demonstration of Increment & Decrement operators in C language has been executed and verified successfully.

F. Demonstration of RELATIONAL OPERATORS.

Aim:

To write a C program for demonstration of Relational operators.

Algorithm:

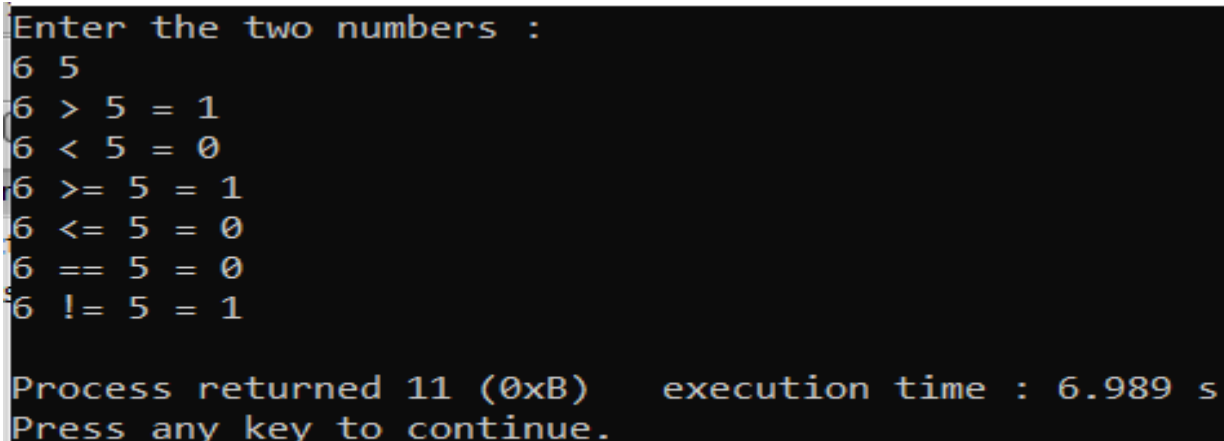
- Step 1: Start
- Step 2: Declare two integers a & b.
- Step 3: Get the values of a & b.
- Step 4: Print the truth value of: $a > b$.
- Step 5: Print the truth value of: $a < b$.
- Step 6: Print the truth value of: $a \geq b$.
- Step 7: Print the truth value of: $a \leq b$.
- Step 8: Print the truth value of: $a == b$.
- Step 8: Print the truth value of: $a != b$.
- Step 9: Stop.

Program:

```
#include<stdio.h>

void main()
{
    int a ,b;
    printf("Enter the two numbers :\n");
    scanf("%d\n%d",&a,&b);
    printf("%d > %d = %d\n",a,b,a>b);
    printf("%d < %d = %d\n",a,b,a<b);
    printf("%d >= %d = %d\n",a,b,a>=b);
    printf("%d <= %d = %d\n",a,b,a<=b);
    printf("%d == %d = %d\n",a,b,a==b);
    printf("%d != %d = %d\n",a,b,a!=b);
}
```

Output:



```
Enter the two numbers :
6 5
6 > 5 = 1
6 < 5 = 0
6 >= 5 = 1
6 <= 5 = 0
6 == 5 = 0
6 != 5 = 1

Process returned 11 (0xB)    execution time : 6.989 s
Press any key to continue.
```

Result:

Thus the program to demonstration of Relational operators in C language has been executed and verified successfully.

G. Demonstration of ASSIGNMENT OPERATORS.

Aim:

To write a C program for demonstration of Assignment operators.

Algorithm:

- Step 1: Start
- Step 2: Declare two integers a & b.
- Step 3: Get the values of a & b.
- Step 4: Calculate the value of a += b and print the value of a,b,a += b.
- Step 5: Calculate the value of a -= b and print the value of a,b,a -= b.
- Step 6: Calculate the value of a /= b and print the value of a,b,a /= b.
- Step 7: Calculate the value of a *= b and print the value of a,b,a *= b.
- Step 8: Calculate the value of a %= b and print the value of a,b,a %= b.
- Step 9: Stop

Program:

```
#include<stdio.h>
void main()
{
    int a,b;
    printf("Enter the two numbers: \n");
    scanf("%d\n%d",&a,&b);
    printf("value of a = %d\n",a);
    printf("value of b = %d\n",b);
    printf("Assigning a = b then :\n new value of a = %d\n",a=b);
    printf("value of a = %d\n",a);
    printf("value of b = %d\n",b);
    printf(" %d += %d == %d\n",a,b,a+=b);
    printf("value of a = %d\n",a);
    printf("value of b = %d\n",b);
    printf(" %d -= %d == %d\n",a,b,a-=b);
    printf("value of a = %d\n",a);
    printf("value of b = %d\n",b);
    printf(" %d *= %d == %d\n",a,b,a*=b);
    printf("value of a = %d\n",a);
    printf("value of b = %d\n",b);
    printf(" %d /= %d == %d\n",a,b,a/=b);
    printf("value of a = %d\n",a);
    printf("value of b = %d\n",b);
    printf(" %d %%= %d == %d\n",a,b,a%=b);
}
```

Output:

```
Enter the two numbers:
6
8
value of a = 6
value of b = 8
Assigning a = b then :
new value of a = 8
value of a = 8
value of b = 8
16 += 8 == 16
value of a = 16
value of b = 8
8 -= 8 == 8
value of a = 8
value of b = 8
64 *= 8 == 64
value of a = 64
value of b = 8
8 /= 8 == 8
value of a = 8
value of b = 8
0 %%= 8 == 0

Process returned 13 (0xD)    execution time : 4.763 s
Press any key to continue.
```

Result:

Thus the program to demonstration of Assignment operators in C language has been executed and verified successfully

H. Demonstration of CONDITIONAL OPERATER.

Aim:

To write a C program for demonstration of Conditional operators.

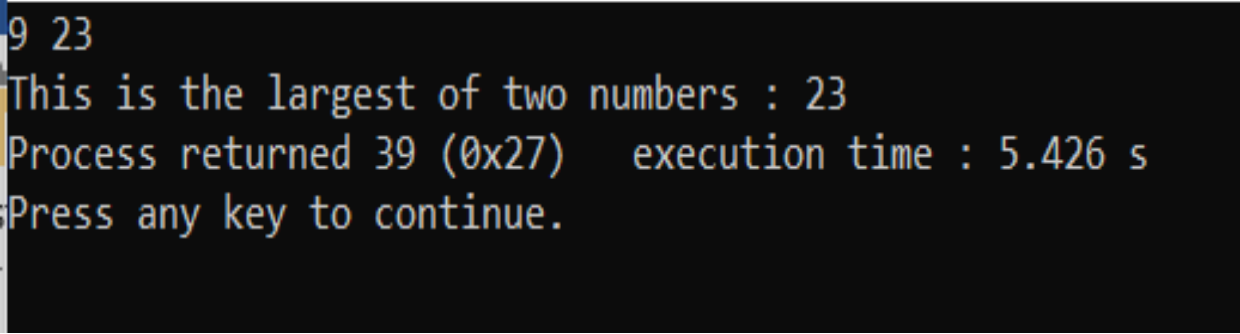
Algorithm:

- Step 1: Start
- Step 2: Declare two integers a , b &c.
- Step 3: Get the values of a & b.
- Step 4: Calculate (a>b) ? a : b.
- Step 5: Store the value of step4 to c.
- Step 6: Print the value of c.
- Step 7: Stop

Program:

```
#include<stdio.h>
void main()
{
    int a,b,c;
    scanf("%d %d",&a,&b);
    c = (a>b)?a:b;
    printf("This is the largest of two numbers : %d",c);
}
```

Output:



```
9 23
This is the largest of two numbers : 23
Process returned 39 (0x27) execution time : 5.426 s
Press any key to continue.
```

Result:

Thus the program to demonstration of Conditional operators in C language has been executed and verified successfully