**Fundamentals of Computer programming**

**ASSIGNMENT**

**III**

**Prepared by: Allen Nembang**

**Shift: Morning**

**Roll Number: 2**

**BscCSIT**

1. **Write A Program to display Sum of two numbers**

**Code:**

|  |
| --- |
| **/\***  **Author: Allen Nembang**  **PROGRAM: To Display Sum of Two Numbers**  **\*/**  **/\***  **Step 1:Start**  **Step 2:Declare variables num1,num2 and sum=0**  **Step 3:Operates sum=num1+num2**  **Step 4:Finally display the value of sum**  **Step 5:Stop**  **\*/**  **#include <stdio.h>**  **#include <stdlib.h>**  **int main()**  **{**  **int num1,num2,sum=0;**  **printf("Enter any two numbers:");**  **scanf("%d%d",&num1,&num2);**  **sum=num1+num2;**  **printf("The sum of two numbers is:%d",sum);**  **return 0;**  **}** |

**Output:**



1. **Write A program to display if the number is positive or negative**

**Code:**

|  |
| --- |
| **/\***  **Author: Allen Nembang**  **PROGRAM: To Display if the Number is Positive or Negative**  **\*/**  **/\***  **Step 1: Start**  **Step 2: Declare variable a**  **Step 3: if a>o**  **Display a is a positive number**  **Else**  **Display a is a negative number**  **Step 4: Stop**  **\*/**  **#include <stdio.h>**  **#include <stdlib.h>**  **int main()**  **{**  **int a;**  **printf("Enter a number: ");**  **scanf("%d", &a);**  **if(a>0)**  **printf("%d is a positive number",a);**  **else**  **printf("%d is a negative number",a);**  **return 0;**  **}** |

**Output:**



1. **Write A program to display if the number is even or odd**

**Code:**

|  |
| --- |
| **/\***  **Author: Allen Nembang**  **PROGRAM: To Display if the Number is even or odd**  **\*/**  **/\***  **Step 1: Start**  **Step 2: Declare a variable a**  **Step 3: if a%2 == 0**  **Display a is an even number**  **Else**  **Display a is an odd number**  **Step 4: Stop**  **\*/**  **#include <stdio.h>**  **#include <stdlib.h>**  **int main()**  **{**  **int a;**  **printf("Enter a number:");**  **scanf("%d", &a);**  **if(a%2 == 0)**  **printf("%d is an even number",a);**  **else**  **printf("%d is an odd number",a);**  **return 0;**  **}** |

**Output:**

****

1. **Write a Program to read three numbers and print the greatest number.**

**Code:**

|  |
| --- |
| **/\***  **Author: Allen Nembang**  **PROGRAM: To Read three numbers and print the greatest number**  **\*/**  **/\***  **Step 1: Start**  **Step 2: Declare variables a, b and c.**  **Step 3: if a>b**  **if a>c**  **Print a is the largest number**  **else**  **Print c is the largest number**  **else**  **if b>c**  **Print b is the largest number**  **else**  **Print c is the largest number**  **Step 4: Stop**  **\*/**  **#include <stdio.h>**  **#include <stdlib.h>**  **int main()**  **{**  **int a,b,c;**  **printf("Enter three numbers:");**  **scanf("%d%d%d",&a,&b,&c);**  **if(a>b){**  **if(a>c){**  **printf("%d is the largest number",a);**  **}else{**  **printf("%d is the largest number",c);**  **}**  **}else{**  **if(b>c){**  **printf("%d is the largest number",b);**  **}else{**  **printf("%d is the largest number",c);**  **}**  **}**  **return 0;**  **}** |

**Output:**

1. **Write a Program to find the sum of the series 1+2+3+4….. Up to entered n numbers.**

**Code:**

|  |
| --- |
| **/\***  **Author: Allen Nembang**  **PROGRAM: To find the sum of the series 1+2+3+4…. Up to entered n numbers**  **\*/**  **/\***  **Step 1: Start**  **Step 2: Declare variables n, i and sum=0.**  **Step 3: Operates sum=( n\*(n+1) ) /2;**  **Step 4: for(i=0; i<=n; i++)**  **if(i !=n)**  **print**  **else**  **print the sum**  **Step 5: Stop**  **\*/**  **#include <stdio.h>**  **#include <stdlib.h>**  **int main()**  **{**  **int n,i;**  **int sum=0;**  **printf("Enter the max values of series: ");**  **scanf("%d",&n);**  **sum = (n \* (n + 1)) / 2;**  **printf("Sum of the series: ");**  **for(i =1;i <= n;i++){**  **if (i!=n)**  **printf("%d + ",i);**  **else**  **printf("%d = %d ",i,sum);**  **}**  **return 0;**  **}** |

**Output:**

1. **Write a Program to display factorial of a given number N**

**Code:**

|  |
| --- |
| **/\***  **Author: Allen Nembang**  **PROGRAM: To Display Factorial of a given number N**  **\*/**  **/\***  **Step 1:Start**  **Step 2:Declare a variable N, fact and i**  **Step 3:Initialize variables fact=1 and i=1**  **Step 4:Read value of N**  **Step 5:Repeat the steps until i=n**  **fact= fact\*i**  **i=i+1**  **Step 6:Display fact**  **Step 7:Stop**  **\*/**  **#include <stdio.h>**  **#include <stdlib.h>**  **int main()**  **{**  **int i,fact=1,num;**  **printf("Enter a number: ");**  **scanf("%d",&num);**  **for(i=1;i<=num;i++)**  **fact=fact\*i;**  **printf("Factorial of %d is: %d",num,fact);**    **return 0;**  **}** |

**Output:**

1. **Write a Program to read 20 numbers and display only sum of even numbers**

**Code:**

|  |
| --- |
| **/\***  **Author: Allen Nembang**  **PROGRAM: To read 20 numbers and display only sum of even numbers**  **\*/**  **/\***  **Step 1: Start**  **Step 2: Declare variables i=2, n and sum=0.**  **Step 3: if(i<=n)**  **sum=sum+i**  **i=i+2**  **Step 4: print sum**  **Step 5: Stop**  **\*/**  **#include <stdio.h>**  **#include <stdlib.h>**  **int main()**  **{**  **int i, n, sum=0;**  **printf("Enter any number: ");**  **scanf("%d", &n);**  **for(i=2; i<=n; i+=2)**  **{**  **sum += i;**  **}**  **printf("\nSum of all even number between 1 to %d = %d", n, sum);**  **return 0;**  **}** |

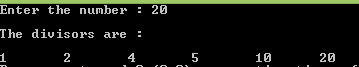
**Output:**

****

1. **Write a Program to read a number n and display all of its divisors.**

**Code:**

|  |
| --- |
| **/\***  **Author: Allen Nembang**  **PROGRAM: To Display Factorial of a given number n**  **\*/**  **/\***  **Step 1:Start**  **Step 2:Declare variables n and i=1**  **Step 3:Read value of n**  **Step 4:Repeat the steps until i=n**  **fact= fact\*i**  **i=i+1**  **Step 5:Display fact**  **Step 6:Stop**  **\*/**  **#include <stdio.h>**  **#include <stdlib.h>**  **int main()**  **{**  **int i, n ;**  **printf("Enter the number : ") ;**  **scanf("%d", &n) ;**  **printf("\nThe divisors are :\n\n") ;**  **for(i = 1 ; i <= n ; i++)**  **if(n % i == 0)**  **printf("%d\t", i) ;**  **return 0;**  **}** |

**Output:**

1. **Write a Program which displays if a number is prime or not.**

**Code:**

|  |
| --- |
| **/\***  **Author: Allen Nembang**  **PROGRAM: To Display if a number is prime or not**  **\*/**  **/\***  **Step 1:Start**  **Step 2:Declare variables a, i=2 and c=0**  **Step 3:Read the value of a**  **Step 4:if (i<a/2) :go to step 6**  **If(a MOD i == 0) then c=1**    **Go to step 6**  **Step 5:i=i+1 go to step 4**  **Step 6:if(c == 0)**  **print a is the prime number**  **else**  **print a is not the prime number**  **Step 7:Stop**  **\*/**  **#include <stdio.h>**  **#include <stdlib.h>**  **int main()**  **{ int a,i,c=0;**  **printf("Enter a number:");**  **scanf("%d",&a);**  **for(i=2;i<a/2;i++){**  **if(a%i==0){**  **c=1;**  **break;**  **}**  **}**  **if(c==0){**  **printf("%d is the prime number",a);**  **}else{**  **printf("%d is not the prime number",a);**  **}**  **return 0;**  **}** |

 **Output:**

1. **What do you mean by Precedence and Associativity of Operators in C. Explain with some examples?**

## Precedence of operators

If more than one operators are involved in an expression, C language has a predefined rule of priority for the operators. This rule of priority of operators is called operator precedence.

In C, precedence of arithmetic operators (\*, %, /, +, -) is higher than relational operators (==,! =, >, <, >=, <=) and precedence of relational operator is higher than logical operators (&&, || and!).

### Example of precedence

(1 > 2 + 3 && 4)

This expression is equivalent to: ((1 > (2 + 3)) && 4)

i.e, (2 + 3) executes first resulting into 5

then, first part of the expression (1 > 5) executes resulting into 0 (false)

then, (0 && 4) executes resulting into 0 (false)

**Output:**

**0**

## Associativity of operators

If two operators of same precedence (priority) is present in an expression, Associativity of operators indicate the order in which they execute.

### Example of associativity

1 == 2 != 3

Here, operators == and != have same precedence. The associativity of both == and != is left to right, i.e, the expression on the left is executed first and moves towards the right.

Thus, the expression above is equivalent to :

((1 == 2) != 3)

i.e, (1 == 2) executes first resulting into 0 (false)

then, (0 != 3) executes resulting into 1 (true)

**Output:1**