**1. Practice sessions**

**AIM a. Write a simple program that prints the results of all the operators available in C (including pre/ post increment , bitwise and/or/not , etc.). Read required operand values from standard input.**

**Arithmetic Operators**

#include <stdio.h>

int main()

{

int a,b,c;

printf(“enter a and b values”);

scanf(“%d %d”,&a,&b);

c = a+b;

printf("a+b = %d \n",c);

c = a-b;

printf("a-b = %d \n",c);

c = a\*b;

printf("a\*b = %d \n",c);

c=a/b;

printf("a/b = %d \n",c);

c=a%b;

printf("Remainder when a divided by b = %d \n",c);

return 0;

}

**Output:**

**Increment and decrement operators**

#include <stdio.h>

int main(){

int a = 10, b = 100;

float c = 10.5, d = 100.5;

printf("a++ = %d \n", a++);

printf("++a = %d \n", ++a);

printf(“b++ = %d \n", b++);

printf("--b = %d \n", --b);

printf("++c = %f \n", ++c);

printf("--d = %f \n", --d);

return 0;

}

**Output**

**Assignment Operators**

#include <stdio.h>

int main()

{

int a = 5, c;

c = a;

printf("c = %d \n", c);

c += a; // c = c+a

printf("c = %d \n", c);

c -= a; // c = c-a

printf("c = %d \n", c);

c \*= a; // c = c\*a

printf("c = %d \n", c);

c /= a; // c = c/a

printf("c = %d \n", c);

c %= a; // c = c%a

printf("c = %d \n", c);

return 0;

}

**Output**

**Relational Operators**

#include <stdio.h>

int main()

{

int a = 5, b = 5, c = 10;

printf("%d == %d = %d \n", a, b, a == b); // true

printf("%d == %d = %d \n", a, c, a == c); // false

printf("%d > %d = %d \n", a, b, a > b); //false

printf("%d > %d = %d \n", a, c, a > c); //false

printf("%d < %d = %d \n", a, b, a < b); //false

printf("%d < %d = %d \n", a, c, a < c); //true

printf("%d != %d = %d \n", a, b, a != b); //false

printf("%d != %d = %d \n", a, c, a != c); //true

printf("%d >= %d = %d \n", a, b, a >= b); //true

printf("%d >= %d = %d \n", a, c, a >= c); //false

printf("%d <= %d = %d \n", a, b, a <= b); //true

printf("%d <= %d = %d \n", a, c, a <= c); //true

return 0;

}

**Output**

**Logical Operators**

#include <stdio.h>

int main()

{

int a = 5, b = 5, c = 10, result;// 1 (true) 0(false)

result = (a == b) && (c > b);

printf("(a == b) && (c > b) equals to %d \n", result);

result = (a == b) && (c < b);

printf("(a == b) && (c < b) equals to %d \n", result);

result = (a == b) || (c < b);

printf("(a == b) || (c < b) equals to %d \n", result);

result = (a != b) || (c < b);

printf("(a != b) || (c < b) equals to %d \n", result);

result = !(a != b);

printf("!(a == b) equals to %d \n", result);

result = !(a == b);

printf("!(a == b) equals to %d \n", result);

return 0;

}

**Output**

**Bitwise operator**

#include <stdio.h>

int main(){

int a = 12, b = 25, num=212, i;

printf("AND = %d", a&b);

printf("OR = %d", a|b);

printf("XOR = %d", a^b);

printf("complement = %d\n",~35);

printf("complement = %d\n",~-12);

for (i=0; i<=2; ++i)

printf("Right shift by %d: %d\n", i, num>>i);

printf("\n");

for (i=0; i<=2; ++i)

printf("Left shift by %d: %d\n", i, num<<i);

return 0;

}

**Output**

**sizeof Operator**

#include <stdio.h>

int main()

{

int a, e[10];

float b;

double c;

char d;

printf("Size of int=%lu bytes\n",sizeof(a));

printf("Size of float=%lu bytes\n",sizeof(b));

printf("Size of double=%lu bytes\n",sizeof(c));

printf("Size of char=%lu byte\n",sizeof(d));

printf("Size of integer type array having 10 elements = %lu bytes\n", sizeof(e));

return 0;

}

**Output**

**Conditional Operator/Ternary Operator**

#include <stdio.h>

int main()

{

char February;

int days;

printf("If this year is leap year, enter 1. If not enter any integer: ");

scanf("%c",&February);

// If test condition (February == 'l') is true, days equal to 29.

// If test condition (February =='l') is false, days equal to 28.

days = (February == '1') ? 29 : 30;

printf("Number of days in February = %d",days);

return 0;

}

**Output**

If this year is leap year, enter 1. If not enter any integer: 1

Number of days in February = 29

**AIM b. Write a simple program that converts one given data type to another using auto conversion and casting. Take the values from standard input.**

**Using auto conversion**

#include<stdio.h>

int main(){

    int x;    // integer x

    char y;  // character c

printf(“enter x and y values”);

scanf(“%d %c”,&x,&y);

    // y implicitly converted to int.

// let y value is ‘a’,ASCII value of 'a' is 97

    x = x + y;

    // x is implicitly converted to float

    float z = x + 1.0;

    printf("x = %d, z = %f", x, z);

    return 0;

}

**Output:**

**Using type Casting**

#include<stdio.h>

 int main()

{

    double x;

printf(“enter x value:”);

  scanf(“%f”,&x);

    // Explicit conversion from double to int

    int sum = (int)x + 1;

     printf("sum = %d", sum);

     return 0;

}

**Output:**