**3. Expression Evaluation**

**a. A building has 10 floors with a floor height of 3 meters each. A ball is dropped from the top of the building. Find the time taken by the ball to reach each floor. (Use the formula s = ut+(1/2)at^2 where u and a are the initial velocity in m/sec (= 0) and acceleration in m/sec^2 (= 9.8 m/s^2)).**

#include<stdio.h>

#include<math.h>

main()

{

int s=30,u=0,r,t;

float a=9.8;

r=2\*s/a;

t=sqrt(r);

printf(“time taken is%d”,t);

}

**b. Write a C program, which takes two integer operands and one operator from the user, performs the operation and then prints the result. (Consider the operators +,-,\*, /, % and use Switch Statement)**

#include <stdio.h>

void main()

{

int a, b, c;

char ch;

printf("Enter your operator(+, -, /, \*, %)\n");

scanf("%c", &ch);

printf("Enter the values of a and b");

scanf("%d%d", &a, &b);

switch(ch)

{

case '+': c = a + b;

printf("addition of two numbers is %d", c);

break;

case '-': c = a - b;

printf("substraction of two numbers is %d", c);

break;

case '\*': c = a \* b;

printf("multiplication of two numbers is %d", c);

break;

case '/': c = a / b;

printf("remainder of two numbers is %d", c);

break;

case '%': c = a % b;

printf("quotient of two numbers is %d", c);

break;

default: printf("Invalid operator");

break;

}

}

**Output:**

+

**c. Write a program that finds if a given number is a prime number.**

#include <stdio.h>

int main()

{

int n, i, flag = 0;

printf("Enter a positive integer: ");

scanf("%d",&n);

for(i=1; i<=n; i++)

{

// condition for nonprime number

if(n%i==0)

{

flag++;

}

}

if (flag==2)

printf("%d is a prime number.",n);

else

printf("%d is not a prime number.",n);

return 0;

}

**Output:**

**d. Write a C program to find the sum of individual digits of a positive integer and test given number is palindrome.**

#include<stdio.h>

main ()

{

int number = 0, digit = 0, sumOfDigits = 0,original\_number,reverse\_integer;

printf("Enter any number");

scanf("%d", &number);

original\_number=number;

while (number != 0)

{

digit = number % 10;

sumOfDigits = sumOfDigits + digit;

reverse\_integer=reverse\_integer\*10 + digit;

number = number / 10;

}

if (original\_number == reverse\_integer)

printf("%d is a palindrome.", original\_number);

else

printf("%d is not a palindrome.", original\_number);

printf ("\nSum of individual digits of a given number is %d", sumOfDigits);

}

**Output:**

**e. A Fibonacci sequence is defined as follows: the first and second terms in the sequence are 0 and 1. Subsequent terms are found by adding the preceding two terms in the sequence. Write a C program to generate the first n terms of the sequence.**

#include <stdio.h>

main()

{

int i, n, t1 = 0, t2 = 1, nextTerm;

printf("Enter the number of terms: ");

scanf("%d", &n);

printf("Fibonacci Series: ");

for (i = 1; i <= n; ++i)

{

printf("%d", t1);

nextTerm = t1 + t2;

t1 = t2;

t2 = nextTerm;

}

return 0;

}

**Output:**

**f. Write a C program to generate all the prime numbers between 1 and n, where n is a value supplied by the user.**

#include<stdio.h>

main()

{

int n, i, j, count;

printf("Enter any number\n");

scanf("%d", &n);

printf("The prime numbers between 1 to %d\n",n);

for(i = 1; i <= n; i++)

{

count = 0;

for(j = 1; j <=i; j++)

{

if(i % j == 0)

{

count++;

}

if(count == 2)

{

printf("%d\t", i);

}

}

}

}

**Output:**

**g. Write a C program to find the roots of a Quadratic equation.**

#include<stdio.h>

#include<math.h>

main()

{

float a,b,c;

float d,root1,root2;

printf("Enter a, b and c of quadratic equation: ");

scanf("%f%f%f",&a,&b,&c);

d = b \* b - 4 \* a \* c;

if(d < 0)

{

printf("Roots are complex number.\n");

printf("Roots of quadratic equation are: ");

printf("%.3f%+.3fi",-b/(2\*a),sqrt(-d)/(2\*a));

printf(", %.3f%+.3fi",-b/(2\*a),-sqrt(-d)/(2\*a));

return 0;

}

else if(d==0)

{

printf("Both roots are equal.\n");

root1 = -b /(2\* a);

printf("Root of quadratic equation is: %.3f ",root1);

return 0;

}

else

{

printf("Roots are real numbers.\n");

root1 = ( -b + sqrt(d)) / (2\* a);

root2 = ( -b - sqrt(d)) / (2\* a);

printf("Roots of quadratic equation are: %.3f , %.3f",root1,root2);

}

return 0;

}

**Output:**

**h. Write a C program to calculate the following, where x is a fractional value. 1-x/2 +x^2/4-x^3/6**

#include<stdio.h>

#include<math.h>

main()

{

float x,s=0,i;

printf(“enter fractional value”);

scanf(“%f”,&x);

for(i=1;i<=3;i++)

{

s+=pow(-x,i)/(2\*i);

}

s=1-s;

printf(“result=%f”,s);

}

**Output:**

**i. Write a C program to read in two numbers, x and n, and then compute the sum of this geometric progression: 1+x+x^2+x^3+………….+x^n. For example: if n is 3 and x is 5, then the program computes 1+5+25+125.**

#include <stdio.h>

#include <math.h>

void main()

{

int n, x, i, sum = 0;

printf("Enter the limit");

scanf("%d", &n);

printf("Enter the value of x");

scanf("%d", &x);

if(x < 0 || n < 0)

{

printf("illegal value");

}

else

{

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

sum=sum + pow(x, i);

}

printf("sum=%d", sum);

}

**Output:**