**Lab Sheet 6**

1. Write a program to add, subtract, multiply and divide two integers using user defined function add(), sub(), mul() and div().

#include <stdio.h>

int add(int a, int b)

{

return a + b;

}

int sub(int a, int b)

{

return a - b;

}

float mul(float a, float b)

{

return a \* b;

}

float div(float a, float b)

{

return a / b;

}

int main( )

{

int x = 10, y = 4;

int a = add(x, y);

int s = sub(x, y);

float m = mul(x, y);

float d = div(x, y);

printf("Addition: %d\n", a);

printf("Subtraction: %d\n", s);

printf("Multiplication: %f\n", m);

printf("Division: %f\n", d);

return 0;

}

1. WAP to display sum of series: x + x2/2! + x3/3! + x4/4! + x5/5! ... xn/n!. User defined function factorial() and power() should be used to calculate the factorial and power.

#include <stdio.h>

int factorial(int n)

{

if (n <= 1)

{

return 1;

}

return n \* factorial(n - 1);

}

int power(int x, int n)

{

if (n == 0)

{

return 1;

}

return x \* power(x, n - 1);

}

int main()

{

float term = 0;

int x, n;

printf("Enter the value of x and n: ");

scanf("%d%d", &x, &n);

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

{

term += power(x, i) / (float)factorial(i);

}

printf("The sum of series: x + x2/2! + x3/3! + x4/4! + x5/5! ... xn/n! s %f", term);

return 0;

}

1. WAP to calculate factorial using Recursion.

#include <stdio.h>

int fact(int n)

{

if (n <= 1)

{

return 1;

}

return n \* fact(n - 1);

}

int main()

{

int n;

printf("Enter a number: ");

scanf("%d", &n);

int factorial = fact(n);

printf("The factorial of %d is : %d.", n, factorial);

return 0;

}

1. WAP to display the nth Fibonacci number using recursion.

#include <stdio.h>

int fibo(int n)

{

if (n == 1 || n == 2)

return 1;

else

return fibo(n - 1) + fibo(n - 2);

}

int main()

{

int n;

printf("Enter the value of n: ");

scanf("%d", &n);

printf("The %dth Fibonacci number is %d\n", n, fibo(n));

return 0;

}

1. WAP to take two numbers in main(). Write a function Swap() to swap the values of the variables. Print the swapped values in main().

#include <stdio.h>

void swap(int \*a, int \*b)

{

int temp = \*a;

\*a = \*b;

\*b = temp;

}

int main()

{

int x, y;

printf("Enter two numbers: ");

scanf("%d%d", &x, &y);

printf("Before swapping:\nx = %d, y = %d\n", x, y);

swap(&x, &y);

printf("After swapping:\nx = %d, y = %d\n", x, y);

return 0;

}

1. WAP to take two float number in main(). Write a function single user define function calculator() to perform the addition, subtraction and multiplication. The sum, difference and product should be displayed from the main() function. [Use the concept of pass by reference.].

#include <stdio.h>

void calc(int x, int y, int \*sum, int \*sub, int \*mul, float \*divide)

{

\*sum = x + y;

\*sub = x - y;

\*mul = x \* y;

\*divide = x / (float)y;

}

int main()

{

int x, y, sum, sub, mul;

float divide;

printf("Enter two numbers: ");

scanf("%d%d", &x, &y);

calc(x, y, &sum, &sub, &mul, ÷);

printf("Sum: %d\n", sum);

printf("Difference: %d\n", sub);

printf("Product: %d\n", mul);

printf("Division: %f\n", divide);

return 0;

}

1. WAP to input a integer number in main(). Write a user define function isPrime() to calculate whether the number is prime of not. Print whether the number is prime or not from the main()

#include <stdio.h>

int isPrime(int num)

{

int flag = 1;

for (int i = 2; i <= num / 2; i++)

{

if (num % i == 0)

{

flag = 0;

}

}

return flag;

}

int main()

{

int a;

printf("Enter a number: ");

scanf("%d", &a);

int prime = isPrime(a);

if (prime == 1)

{

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

}

else

{

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

}

return 0;

}

1. WAP to illustrate the concept of global and static variables.

#include <stdio.h>

// declaring global variable;

int global = 5;

void change()

{

global = 15;

}

int main()

{

// global variable

printf("Global variable: %d\n", global);

global = 10;

printf("Global variable after changing from main: %d\n", global);

change();

printf("Global variable after changing from change(): %d\n", global);

// static variable

for (int i = 0; i < 5; i++)

{

static int count = 0;

count++;

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

}

return 0;

}