**Pointer and Function**

**Q1 - WAP to accept an array of n integers and calculate**

**sum of odd numbers and even numbers using the pointer to an array.**

#include <stdio.h>

int main()

{

int i,e=0,o=0,n;

int \*ptr;

printf("Enter the array element : ");

scanf("%d",&n);

int a[n];

printf("Enter Numbers:\n");

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

scanf("%d",&a[i]);

ptr= a;

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

{

if(\*ptr%2==0)

e=e+\*ptr;

else

o=o+\*ptr;

ptr++;

}

printf("\nSum of Even Numbers = %d",e);

printf("\nSum of Odd Numbers = %d",o);

return 0;

}

**Output**

Enter the array element : 3

Enter Numbers: 1 2 3

Sum of Even Numbers = 2

Sum of Odd Numbers = 4

**Q2 - Write a function isEven, which accepts an integer as parameter and returns 1 if the number is even, and 0 otherwise. Use this function in main to accept n numbers and ckeck if they are even or odd.**

#include<stdio.h>

int main(){

int n;

printf("Enter the number to check even or not :");

scanf("%d",&n);

int ans = isEven(n);

printf("%d",ans);

return 0;

}

int isEven(n){

if (n%2==0)

{

return 1;

}

else

{

return 0;

}

}

**Output**

Enter the number to check even or not : 3

0

**Q3 - Write a function isPrime, which accepts an integer as parameter and returns 1 if the number is prime and 0 otherwise. Use this function in main to display the first 10 prime numbers.**

#include<stdio.h>

int main(){

int n;

printf("Enter the number to check prime or not :");

scanf("%d",&n);

int ans = isPrime(n);

printf("%d",ans);

return 0;

}

int isPrime(n){

printf("Prime numbers are:-\n");

for(int i=2;i<n;i++)

{

int c=0;

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

{

if(i%j==0)

{

c++;

}

}

if(c==2)

{

printf("%d ",i);

}

}

}

**Output**

Enter the number to check prime or not :10

Prime numbers are:- 2 3 5 7

**Q4 -For the following standard functions, write corresponding user defined functions and**

**write a menu driven program to use them. strcat, strcmp, strrev, strupr.**

#include<stdio.h>

#include<string.h>

int main(){

int n;

char a[100],b[100];

printf("\n.......Menu......\n");

printf("1 - String concatanation \n");

printf("2 - string compaire \n");

printf("3 - string reverse \n");

printf("4 - String convert to Uppercase \n");

scanf("%d",&n);

switch (n)

{

case 1:

printf("Enter the first string \n");

scanf("%s",&a);

printf("Enter the second string \n");

scanf("%s",&b);

strCon(a , b);

printf("After concatation string is : %s ",a);

break;

case 2:

printf("Enter the first string \n");

scanf("%s",&a);

printf("Enter the second string \n");

scanf("%s",&b);

strComp(a , b);

break;

case 3:

printf("Enter the first string \n");

scanf("%s",&a);

StringRev(a);

printf("Reverse of string A is : %s",a);

break;

case 4:

printf("Enter the first string \n");

scanf("%s",&a);

strUpper(a);

printf("Uppercase of string A is : %s",a);

break;

}

return 0;

}

void strCon(char a[100] , char b[100])

{

int i , len = 0;

for ( i = 0; i < a[i]!='\0'; i++)

{

len++;

}

for ( i = 0; i < b[i]!='\0'; i++)

{

a[len+i]=b[i];

}

a[len+i]='\0';

}

void strComp(char a[100],char b[100])

{

int i, len1=0,len2=0;

for (i = 0; i < a[i]!='\0'; i++)

{

len1++;

}

for (i = 0; i < b[i]!='\0'; i++)

{

len2++;

}

if(len1==len2)

{

printf("String is same ");

}

else

{

printf("String is not same");

}

}

void StringRev(char a[100])

{

int i, len=0,left,right;

for (i = 0; i < a[i]!='\0'; i++)

{

len++;

}

left=0;

right=len-1;

for ( i = left; i < right; i++)

{

int temp=a[i];

a[i]=a[right];

a[right]=temp;

}

}

void strUpper(char a[100])

{

int i;

for ( i = 0; i < a[i]!='\0'; i++)

{

if(a[i]>=97&&a[i]<=122)

{

a[i]=a[i]-32;

}

}

}

**Output**

.......Menu......

1 - String concatanation

2 - string compaire

3 - string reverse

4 - String convert to Uppercase

4

Enter the first string

Mahi

**Q5 - Write a function power, which calculates x raised to y. Write another function, which calculates n! Using for loop. Use these functions to calculate the sum of first n terms of the Taylor series:**

sin(x) = x - 3! 3 x + 5! 5 x + ……

#include<stdio.h>

#include<math.h>

int main()

{

cal();

}

void cal()

{

int x,i;

int fact = 1,n;

float sum=0;

printf("\nEnter the value of x in the series : ");

scanf("%d",&x);

printf("\nEnter the number of terms in the series : ");

scanf("%d",&n);

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

{

fact = fact\*i;

sum = sum + (power(x,i)/fact) ;

}

sum= sum +1;

printf("\nThe sum of the taylor series is : %.2f\n\n",sum);

}

float power(int x, unsigned int i)

{

if (i == 0)

return 1;

else if (i%2 == 0)

return power(x, i/2)\*power(x, i/2);

else

return x\*power(x, i/2)\*power(x, i/2);

}

**Output**

Enter the value of x in the series : 4

Enter the number of terms in the series : 6

The sum of the taylor series is : 41.00

**Q6 - Write a recursive C function to calculate the GCD of two numbers. Use this function in main.**

**The GCD is calculated as : gcd(a,b) = a if b = 0 = gcd (b, a mod b) otherwise.**

#include<stdio.h>

int main(){

int num1 ,num2;

printf("Enter the first number = ");

scanf("%d",&num1);

printf("Enter the second number = ");

scanf("%d",&num2);

int ans = GCD(num1,num2);

printf("GCD of this two number is :%d",ans);

return 0;

}

int GCD(int num1, int num2)

{

int GCD;

for (int i = 1; i<=num1&&i<=num2; i++)

{

if (num1%i==0&&num2%i==0)

{

GCD=i;

}

}

return GCD;

}

**Output**

Enter the first number = 49

Enter the second number = 35

GCD of this two number is :7

**Q7 - Write a recursive C function to calculate x raised to y . (Do not use standard library function)**

#include<stdio.h>

int main(){

int y ,x;

printf("Enter the value of base number = X : ");

scanf("%d",&x);

printf("Enter the value of power = y: ");

scanf("%d",&y);

int ans= power(x,y);

printf("%d Raised to power %d is = %d",x,y,ans);

return 0;

}

int power(int b,int e)

{

if(e==0)

return 1;

else

return (b\*power(b,e-1));

}

**Output**

Enter the value of base number = X : 5

Enter the value of power = y: 2

5 Raised to power 2 is = 25

**Q8 - Write a recursive function to calculate the sum of digits of a number till you get a single digit number.Example: 961 -> 16 -> 5. (Note: Do not use a loop)**

#include<stdio.h>

int main(){

int x;

printf("Enter the value of X to perform digit operation : ");

scanf("%d",&x);

int ans = SingleDigit(x);

printf("Sum of digit is : %d",ans);

return 0;

}

int SingleDigit(int a)

{

if (a != 0)

{

return (a % 10 + SingleDigit (a / 10));

}

else

{

return 0;

}

}

**Output**

Enter the value of X to perform digit operation : 123

Sum of digit is : 6

**Q9 - Write a recursive function to calculate the nth Fibonacci number. Use this function in main to**

**display the first n Fibonacci numbers.The recursive definition of nth Fibonacci number is as follows:**

**fib(n) = 0 if n = 1 = 1 if n = 2 = fib(n-2) + fib(n-1) if n>2**

#include<stdio.h>

int main(){

int n;

printf("Enter the Number for calculating th fibonacci number :");

scanf("%d",&n);

int ans = fib(n);

printf("%d th number of Fibonacci series is :%d \n",n,ans);

return 0;

}

int fib(int x)

{

if (x==1)

{

return 0;

}

else if (x==2)

{

return 1;

}

if (x>2)

{

return (fib(x-2) + fib(x-1));

}

}

**Output**

Enter the Number for calculating th fibonacci number :8

8 th number of Fibonacci series is :13