Ex 0 :

// call a function using pointer

#include <stdio.h>

void add(int x, int y)

{

printf("first value %d",x);

printf("second value %d",y);

printf("addition = %d",x + y);

}

int main(void)

{

int a, b;

void (\*ptr)(int, int);

ptr = add;

printf("enter first value: ");

scanf("%d",&a);

printf("enter Second value: ");

scanf("%d",&b);

ptr(a,b);

return 0;

}

Out put : enter first value: enter Second value: first value 4second value 5addition = 9

====================================

Ex 1.

#include <stdio.h>

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

{

int temp = \*a; \*a = \*b;

\*b = temp;

}

int gcf(int a, int b)

{

if (b > a) swap(&a, &b);

while (b) {

int temp = b ;

b = a % b ;

a = temp ;

}

return a;

}

int main(void)

{

int a = 3, b = 5;

printf("GCF of %d and %d is %d\n", a, b, gcf(a, b) );

return 0;

// your code goes here

return 0;

}

Output :

GCF of 3 and 5 is 1

Ex2

// DOES NOT WORK AS EXPECTED..

#include <stdio.h>

void swap(int a, int b)

{

int temp = a;

a = b;

b = temp;

}

int main(void)

{

int a = 3, b = 5;

swap(a, b);

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

return 0;

}

===================== output ======================

a=3, b=5

// DOES WORK AS EXPECTED :

EX 3

#include <stdio.h>

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

{

int temp = \*a;

\*a = \*b;

\*b = temp;

}

int main(void)

{

int a = 3, b = 5;

swap(&a, &b);

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

return 0;

}

==== OUT PUT ========================

a=5, b=3

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\*\*\*\*\*\*\*\*\*\*\*\*\*example 4 **Program to convert binary number to decimal**

**#include <stdio.h>**

**#include <math.h>**

**int convertBinaryToDecimal(long long n);**

**int main()**

**{**

**long long n;**

**printf("Enter a binary number: ");**

**scanf("%lld", &n);**

**printf("%lld in binary = %d in decimal", n, convertBinaryToDecimal(n));**

**return 0;**

**}**

**int convertBinaryToDecimal(long long n)**

**{**

**int decimalNumber = 0, i = 0, remainder;**

**while (n!=0)**

**{**

**remainder = n%10;**

**n /= 10;**

**decimalNumber += remainder\*pow(2,i);**

**++i;**

**}**

**return decimalNumber;**

**}**

**=========== output ===========================**

**Enter a binary number: 110110111**

**110110111 in binary = 439**