# ASSIGNMENT 1.FOCP

//Q1. WAP to check whether a given is Armstrong or not ?

#include <stdio.h>

int main()

{

int num,sum = 0,temp,digit;

printf("enter an integer :");

scanf("%d", &num );

temp=num ;

while( temp>0){

digit=temp%10;

sum += digit \* digit \*digit;

temp/=10;

}

if (sum==num){

printf("%d is an Armstrong \n", num);

}

else{

printf("%d is not Armstrong \n",num);

}return 0;

}

//Q2. WAP to read two integers and print hcf?

#include <stdio.h>

int main()

{

int a,b,hcf,i;

printf("enter a and b :");

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

for(i=1;i<=a && i<+b;i++){

if(a%i==0 && b%i==0){

hcf = i;

}}

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

return 0;

}

//Q3. WAP to subtract two integers without using Minus (-) operator. (Hint Bitwise operator)

#include <stdio.h>

int main()

{

int a,b;

printf("enter a and b :");

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

int sub;

sub=~b;

sub=sub+1;

sub=sub+a;

printf("sub of %d and %d is %d", a,b,sub);

return 0;

}

METHOD 1:

//Q4.WAP to accept two numbers and swap them

#include <stdio.h>

int main()

{

int a,b,c;

printf("enter a and b :");

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

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

c=a;

a=b;

b=c;

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

return 0;

}

METHOD 2:

//Q4.WAP to accept two numbers and swap them

#include <stdio.h>

int main()

{

int a,b,c;

printf("enter a and b :");

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

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

a=a+b;

b=a-b;

a=a-b;

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

return 0;

}

METHOD 3:

//Q4.WAP to accept two numbers and swap them

#include <stdio.h>

int main()

{

int a,b,c;

printf("enter a and b :");

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

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

a=a^b;

b=a^b;

a=a^b;

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

return 0;

}

METHOD 4:

#include<stdio.h>

void swap(int \*x,int \*y){

int temp;

temp=\*x;

\*x=\*y;

\*y=temp;

}

int main(){

int a,b;

printf("enter first integer(a) : ");

scanf("%d", &a);

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

swap(&a,&b);

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

}

Q5. WAP to check whethernumberis Perfect Numberornot. (To check perfect number, wehaveto find all divisors of that number and find their sum, if sum of divisors is equal to number it means numberis Perfect Number.)  
#include <stdio.h>

int isperfectnumber(int num)

{

int sum =0;

for (int i=1; i<=num/2; i++)

{

if (num % i == 0)

{

sum ==i;

} }

return sum == num;

}

int main () {

int number;

printf("enter a number: ");

scanf ("%d",&number);

if (isperfectnumber(number))

{

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

}

else{

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

}

return 0;

}

Q.6 . WAP to accept a coordinate point in an XY coordinate system and determine in which quadrant the coordinate point lies.

#include <stdio.h>

int main(){

int x, y;

printf("Enter the value for x and y: ");

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

if (x > 0 && y > 0){

printf("This point lies in the first quadrant.");}

else if (x < 0 && y > 0){

printf("This point lies in the second quadrant.");}

else if (x < 0 && y < 0){

printf("This point lies in the third quadrant.");}

else if (x > 0 && y < 0){

printf("This point lies in the fourth quadrant.");}

else if (x == 0 && y == 0){

printf("This point lies at the origin.");}

else if (x==0){

printf("The point lies at y-axis");

}

else if (y==0){

printf("The point lies at x-axis");

}

return 0;

}

Q7.WAP to binary to decimal conversion $ decimal to binary for a given number as per users choice?

#include <stdio.h>

int binaryTodecimal(int binary)

{

int decimal= 0,base =1,rem;

while (binary > 0){

rem = binary %10;

decimal == rem \*base;

binary /=10;

base \*= 2;

}

return decimal;

}

void decimalTobinary(int decimal){

if (decimal == 0){

printf("0");

return;

}

int binary[32],index=0;

while(decimal>0){

binary[index++]=decimal %2;

decimal /=2;

}

for(int i=index-1;i>=0;i--)

printf("%d",binary[i]);

}

int main(){

int choice,num;

printf("1.binary to decimal\n2.decimalTobinary\nchoose: ");

scanf("%d",&choice);

if(choice==1){

printf("enter binary :");

scanf("%d",&num);

printf("decimal : %d\n",binaryTodecimal(num));

}

else if(choice==2){

printf("enter decimal :");

scanf("%d",&num);

printf("binary :");

decimalTobinary(num);

printf("\n");

}else{

printf("invalid choice.\n");

}

}

Q.8. WAP to print below mentioned pattern:

#include <stdio.h>

int main ()

{

int rows = 5;

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

{

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

{

if ((i+j) % 2 == 0)

{

printf("1");}

else{

printf("0");

}}

printf ("\n");

}

return 0;

}

Q.9. Wap to print following pyramid:

#include <stdio.h>

int main ()

{

int rows = 5;

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

{

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

{

printf("%d",j % 2);

}

for (int j= 0; j< (rows -i -1) \* 2; j++)

{ printf (" ");

}

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

{ printf ("%d",j % 2);

}

printf("\n");

}

for (int i = 0;i< rows \* 2; i++)

{

printf("%d",i % 2);}

printf ("\n");

return 0;

}

Q.10. WAP to print Pascal’s Triangle:

#include <stdio.h>

int main ()

{

int rows ,coef= 1;

printf("enter the number of rows: ");

scanf ("%d",&rows);

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

{

for (int j = 0;j < rows - i;j++)

{

printf (" ");

}

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

{ if (j == 0 || i == 0)

{

coef = 1;

}

else{

coef = coef \* (i - j + 1) / j;

}

printf("%d",coef);

}

printf ("\n");

}

return 0;

}