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

#include<stdio.h>

int main(){

int num,originalnum,remainder,result=0;

printf("enter a three digit number:\n");

scanf("%d",&num);

originalnum=num;

while(originalnum!=0){

remainder=originalnum%10;

result+=remainder\*remainder\*remainder;

originalnum/=10;

}

if(result==num)

printf("%d num is armstrong",num);

else

printf("%d num is not armstrong",num);

return 0;

}

Q2. WAP to read two integers and print their HCF (Highest Common Factor).

#include<stdio.h>

int main(){

int num1,num2,i,hcf;

printf("enter two integer");

scanf("%d%d",&num1,&num2);

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

{

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

hcf=i;

}

printf("hcf of %d and %d is %d",num1,num2,hcf);

return 0;

}

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

#include<stdio.h>

int main()

{

int a, b, sub;

printf("Enter 2 numbers\n");

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

sub = a+~b+1;

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

return 0;

}  
Q4. WAP to accept two integer numbers and swap them using 4 different methods in C  
language.

Method 1:

#include <stdio.h>

int main(){

int a,b,c;

printf("provide two integer:");

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

c=a;

a=b;

b=c;

printf("after swaping a=%dand b=%d",a,b);

return 0;

}

Method 2:

#include <stdio.h>

int main(){

int a,b;

printf("provide two integer:");

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

a=b-a;

b=b-a;

a=b+a;

printf("after swaping a=%dand b=%d",a,b);

return 0;

}

Method 3:

#include <stdio.h>

int main() {

int a, b;

printf("Enter two integers: ");

scanf("%d %d", &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>

int main() {

int a, b;

printf("Enter two integers: ");

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

a = a \* b;

b = a / b;

a = a / b;

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

return 0;

}  
Q5. WAP to check whether number is Perfect Number or not.  
#include<stdio.h>

int main(){

int number,i,result=0;

printf("enter the number:\n");

scanf("%d",&number);

for(i=1;i<=number;i++){

if(number%i==0)

result=result+i;

}

if(result==2\*number)

printf("perfect number\n");

else

printf("not perfect number");

}  
Q6. WAP to accept a coordinate point in an XY coordinate system and determine in which  
quadrant the coordinate point lies.  
Test Data: 7 9  
Expected Output: The coordinate point (7,9) lies in the First quadrant.

#include <stdio.h>

int main(){

int x,y;

printf("provide x coordinate");

scanf("%d",&x);

printf("provide y coordinate");

scanf("%d",&y);

printf("\n");

if(x>0&&y>0)

printf("the coordinate point (%d,%d) lies in first quadrant",x,y);

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

printf("the coordinate point (%d,%d) lies in second quadrant",x,y);

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

printf("the coordinate point (%d,%d) lies in third quadrant",x,y);

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

printf("the coordinate point (%d,%d) lies in fourth quadrant",x,y);

return 0;

}  
Q7. WAP for Binary to Decimal conversion & Decimal to Binary for a given number as per  
user’s choice.

include <stdio.h>

int main() {

int i = 0, rem, n, a[10], c;

int decimal = 0, b = 1;

printf("Press 1 for decimal to binary and 2 for binary to decimal: ");

scanf("%d", &c);

switch(c) {

case 1:

printf("Provide number: ");

scanf("%d", &n);

while (n) {

rem = n % 2;

n = n / 2;

a[i] = rem;

i++;

}

printf("Binary representation: ");

for (int j = i - 1; j >= 0; j--) {

printf("%d", a[j]);

}

printf("\n");

break;

case 2:

printf("Provide binary number: ");

scanf("%d", &n);

while (n != 0) {

rem = n % 10;

decimal = decimal + rem \* b;

n = n / 10;

b = b \* 2;

}

printf("The decimal number is: %d\n", decimal);

break;

default:

printf("Invalid choice.\n");

}

return 0;

}  
Q8. WAP to print below mentioned pattern:  
1  
01  
101  
0101  
10101

#include <stdio.h>

int main(){

int i,j;

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

{

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

{

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

{printf("1");}

else

{printf("0");}

}

printf("\n");

}

return 0;

}  
Q9. WAP to print following Pyramid:  
0 0  
01 01  
010 010  
0101 0101  
0101001010  
Q10. WAP to print Pascal’s Triangle.

#include <stdio.h>

int main() {

int n;

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

scanf("%d", &n);

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

int value = 1;

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

printf("%d ", value);

value=value\*(i-j)/j;

}

printf("\n");

}

return 0;

}