**C Program to Display Prime Numbers Between Intervals Using User-defined Function**

#include<stdio.h>

int check\_prime(int num);

int main(){

int n1,n2,i,flag;

printf("Enter two numbers(intervals): ");

scanf("%d %d",&n1, &n2);

printf("Prime numbers between %d and %d are: ", n1, n2);

for(i=n1+1;i<n2;++i)

{

flag=check\_prime(i);

if(flag==0)

printf("%d ",i);

}

return 0;

}

int check\_prime(int num) /\* User-defined function to check prime number\*/

{

int j,flag=0;

for(j=2;j<=num/2;++j){

if(num%j==0){

flag=1;

break;

}

}

return flag;

}

**C Program to Check Prime and Armstrong Number by Making Function**

**(Armstrong Numbers: 0,1,153,370,371,407 etc)**

/\* C program to check either prime number or Armstrong number depending upon the data entered by user. \*/

#include <stdio.h>

int prime(int n);

int armstrong(int n);

int main()

{

char c;

int n,temp=0;

printf("Eneter a positive integer: ");

scanf("%d",&n);

printf("Enter P to check prime and A to check Armstrong number: ");

c=getche();

if (c=='p' || c=='P')

{

temp=prime(n);

if(temp==1)

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

else

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

}

if (c=='a' || c=='A')

{

temp=armstrong(n);

if(temp==1)

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

else

printf("\n%d is not an Armstrong number.",n);

}

return 0;

}

int prime(int n)

{

int i, flag=1;

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

{

if(n%i==0)

{

flag=0;

break;

}

}

return flag;

}

int armstrong(int n)

{

int num=0, temp, flag=0;

temp=n;

while(n!=0)

{

num+=(n%10)\*(n%10)\*(n%10);

n/=10;

}

if (num==temp)

flag=1;

return flag;

}

**C program to Check Whether a Number can be Express as Sum of Two Prime Numbers**

#include <stdio.h>

int prime(int n);

int main()

{

int n, i, flag=0;

printf("Enter a positive integer: ");

scanf("%d",&n);

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

{

if (prime(i)!=0)

{

if ( prime(n-i)!=0)

{

printf("%d = %d + %d\n", n, i, n-i);

flag=1;

}

}

}

if (flag==0)

printf("%d can't be expressed as sum of two prime numbers.",n);

return 0;

}

int prime(int n) /\* Function to check prime number \*/

{

int i, flag=1;

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

if(n%i==0)

flag=0;

return flag;

}

**C Program to Convert Binary Number to Decimal and Decimal to Binary**

/\* C programming source code to convert either binary to decimal or decimal to binary according to data entered by user. \*/

#include <stdio.h>

#include <math.h>

int binary\_decimal(int n);

int decimal\_binary(int n);

int main()

{

int n;

char c;

printf("Instructions:\n");

printf("1. Enter alphabet 'd' to convert binary to decimal.\n");

printf("2. Enter alphabet 'b' to convert decimal to binary.\n");

scanf("%c",&c);

if (c =='d' || c == 'D')

{

printf("Enter a binary number: ");

scanf("%d", &n);

printf("%d in binary = %d in decimal", n, binary\_decimal(n));

}

if (c =='b' || c == 'B')

{

printf("Enter a decimal number: ");

scanf("%d", &n);

printf("%d in decimal = %d in binary", n, decimal\_binary(n));

}

return 0;

}

int decimal\_binary(int n) /\* Function to convert decimal to binary.\*/

{

int rem, i=1, binary=0;

while (n!=0)

{

rem=n%2;

n/=2;

binary+=rem\*i;

i\*=10;

}

return binary;

}

int binary\_decimal(int n) /\* Function to convert binary to decimal.\*/

{

int decimal=0, i=0, rem;

while (n!=0)

{

rem = n%10;

n/=10;

decimal += rem\*pow(2,i);

++i;

}

return decimal;

}

**C Program to Convert Octal Number to Decimal and Decimal to Octal**

/\* C programming source code to convert either octal to decimal or decimal to octal according to data entered by user. \*/

#include <stdio.h>

#include <math.h>

int decimal\_octal(int n);

int octal\_deciaml(int n);

int main()

{

int n;

char c;

printf("Instructions:\n");

printf("1. Enter alphabet 'o' to convert decimal to octal.\n");

printf("2. Enter alphabet 'd' to convert octal to decimal.\n");

scanf("%c",&c);

if (c =='d' || c == 'D')

{

printf("Enter an octal number: ");

scanf("%d", &n);

printf("%d in octal = %d in decimal", n, octal\_decimal(n));

}

if (c =='o' || c == 'O')

{

printf("Enter a decimal number: ");

scanf("%d", &n);

printf("%d in decimal = %d in octal", n, decimal\_octal(n));

}

return 0;

}

int decimal\_octal(int n) /\* Function to convert decimal to octal \*/

{

int rem, i=1, octal=0;

while (n!=0)

{

rem=n%8;

n/=8;

octal+=rem\*i;

i\*=10;

}

return octal;

}

int octal\_decimal(int n) /\* Function to convert octal to decimal \*/

{

int decimal=0, i=0, rem;

while (n!=0)

{

rem = n%10;

n/=10;

decimal += rem\*pow(8,i);

++i;

}

return decimal;

}

**C Program to Convert Binary Number to Octal and Octal to Binary**

/\* C programming source code to convert either binary to octal or octal to binary according to data entered by user. \*/

#include <stdio.h>

#include <math.h>

int binary\_octal(int n);

int octal\_binary(int n);

int main()

{

int n;

char c;

printf("Instructions:\n");

printf("1. Enter alphabet 'o' to convert binary to octal.\n");

printf("2. Enter alphabet 'b' to convert octal to binary.\n");

scanf("%c",&c);

if ( c=='o' || c=='O')

{

printf("Enter a binary number: ");

scanf("%d",&n);

printf("%d in binary = %d in octal", n, binary\_octal(n));

}

if ( c=='b' || c=='B')

{

printf("Enter a octal number: ");

scanf("%d",&n);

printf("%d in octal = %d in binary",n, octal\_binary(n));

}

return 0;

}

int binary\_octal(int n) /\* Function to convert binary to octal. \*/

{

int octal=0, decimal=0, i=0;

while(n!=0)

{

decimal+=(n%10)\*pow(2,i);

++i;

n/=10;

}

/\*At this point, the decimal variable contains corresponding decimal value of binary number. \*/

i=1;

while (decimal!=0)

{

octal+=(decimal%8)\*i;

decimal/=8;

i\*=10;

}

return octal;

}

int octal\_binary(int n) /\* Function to convert octal to binary.\*/

{

int decimal=0, binary=0, i=0;

while (n!=0)

{

decimal+=(n%10)\*pow(8,i);

++i;

n/=10;

}

/\* At this point, the decimal variable contains corresponding decimal value of that octal number. \*/

i=1;

while(decimal!=0)

{

binary+=(decimal%2)\*i;

decimal/=2;

i\*=10;

}

return binary;

}