#include<stdio.h> #include<stdlib.h> #include<limits.h>

long long binomial\_coeff(long long m[][200],int n,int k);void calculate(int n);long long DP\_binomial\_coeff(int n,int k);

int main()

{

int num;printf("Enter the no. to be checked for primality:");

scanf("%d",&num); //input a no.

if(num<=1){printf("Invalid input");exit(0);} //checking for invalid input

calculate(num);

return 0;}

void calculate(int num)

{

int check,i,j,f;long long nCk;

if(num%2==0) //using if else in order to calculate the limit upto which binomial coefficients should be calculated

check=(num/2)+1;

else

check=(num+1)/2;

if(check>0){printf("\nCoefficients to be checked for divisibility are:\n");}

for(i=2;i<check;i++)

{

nCk=DP\_binomial\_coeff(num,i); //calculating the value of binomial coefficents

if(nCk%num!=0) //checking the divisiblity of binomial coefficients by the input value

{

if(nCk<=0)

{printf("\n%dC%d is GREATER THAN : 600851475143 (The upper bound of long long)\nSo can't Check any further....\nsince this a rudimentary model of AKS primality test!!",num,i);exit(0);}

printf("%llu is not divisible by %d\n",nCk,num);printf("\nThe number %d is composite.",num);exit(0);} //if any coefficient is not divisible by the no.,then it is composite

else

printf("%llu is divisible by %d\n",nCk,num);

}printf("\nThe number %d is prime!",num);}

long long DP\_binomial\_coeff(int n,int k) //if all are divisible,then it is prime

{

int i,j;long long m[n][200];

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

{for(j=0;j<200;j++)

m[i][j]=0;}

return binomial\_coeff(m,n,k);

}

long long binomial\_coeff(long long m[][200],int n,int k) //calculating the value of binomial coefficients using dynamic programming top down approach

{if(k==0 || k==n)

return 1;

if(m[n-1][k-1]!=0)

return m[n-1][k-1];

else

{m[n-1][k-1]=binomial\_coeff(m,n-1,k-1)+binomial\_coeff(m,n-1,k);

return m[n-1][k-1]; }}