#include <iostream>

using namespace std;

/\* run this program using the console pauser or add your own getch, system("pause") or input loop \*/

int main(int argc, char\*\* argv) {

int CEyear, CHyear;

cout<<"請輸入民國年";

cin>>CHyear;

CEyear=CHyear+1911;

if(CEyear%4!=0){

cout<<CEyear<<"是平年"<<endl;}

else if (CEyear%100!=0){

cout<<CEyear<<"是閏年"<<endl;}

else if (CEyear%400!=0){

cout<<CEyear<<"是平年"<<endl;}

else{ //是4 100 400 倍數

cout<<CEyear<<"是閏年"<<endl;}

int year, week, day, Februrar, firstday, month, daybefore; //firstday is the weekend of 1/1

cout<<"Input the year and the weekend of 1.1"<<endl;

cin>>year>>firstday;

cout<<"Input the month and the day";

cin>>month>>day;

if(year%4 ==0 && year%100 != 0 || year%400 ==0)

{Februrar=29; //閏年 //daybefore is one month before the moment month ex: March, daybefore is 31+29

if (month==1)

{

daybefore=0;}

if (month==2)

{

daybefore=31;}

if (month==3)

{

daybefore=31+29;}

if (month==4)

{

daybefore=31+29+31;}

if (month==5)

{

daybefore=31+29+31+30;}

if (month==6)

{

daybefore=31+29+31+30+31;}

if (month==7)

{

daybefore=31+29+31+30+31+30;}

if (month==8)

{

daybefore=31+29+31+30+31+30+31;}

if (month==9)

{

daybefore=31+29+31+30+31+30+31+31;}

if (month==10)

{

daybefore=31+29+31+30+31+30+31+31+30;}

if (month==11)

{

daybefore=31+29+31+30+31+30+31+31+30+31;}

if (month==12)

{

daybefore=31+29+31+30+31+30+31+31+30+31+30;}

if( (daybefore + firstday + day-1)%7==1)

{cout<<"Monday";}

if( (daybefore + firstday + day-1)%7==2)

{cout<<"Tuesday";}

if( (daybefore + firstday + day-1)%7==3)

{cout<<"Wednesday";}

if( (daybefore + firstday + day-1)%7==4)

{cout<<"Thursday";}

if( (daybefore + firstday + day-1)%7==5)

{cout<<"Friday";}

if( (daybefore + firstday + day-1)%7==6)

{cout<<"Saturday";}

if( (daybefore + firstday + day-1)%7==0)

{cout<<"Sunday";}

}

else {

Februrar=28;

if (month==1)

{

daybefore=0;}

if (month==2)

{

daybefore=31;}

if (month==3)

{

daybefore=31+28;}

if (month==4)

{

daybefore=31+28+31;}

if (month==5)

{

daybefore=31+28+31+30;}

if (month==6)

{

daybefore=31+28+31+30+31;}

if (month==7)

{

daybefore=31+28+31+30+31+30;}

if (month==8)

{

daybefore=31+28+31+30+31+30+31;}

if (month==9)

{

daybefore=31+28+31+30+31+30+31+31;}

if (month==10)

{

daybefore=31+28+31+30+31+30+31+31+30;}

if (month==11)

{

daybefore=31+28+31+30+31+30+31+31+30+31;}

if (month==12)

{

daybefore=31+28+31+30+31+30+31+31+30+31+30;}

if( (daybefore + firstday + day-1)%7==1)

{cout<<"Monday";}

if( (daybefore + firstday + day-1)%7==2)

{cout<<"Tuesday";}

if( (daybefore + firstday + day-1)%7==3)

{cout<<"Wednesday";}

if( (daybefore + firstday + day-1)%7==4)

{cout<<"Thursday";}

if( (daybefore + firstday + day-1)%7==5)

{cout<<"Friday";}

if( (daybefore + firstday + day-1)%7==6)

{cout<<"Saturday";}

if( (daybefore + firstday + day-1)%7==0)

{cout<<"Sunday";}

}

return 0;

}

#include <iostream>

#include <cmath>

using namespace std;

/\* run this program using the console pauser or add your own getch, system("pause") or input loop \*/

int main(int argc, char\*\* argv) {

float a,b,c,L1,L2,L1real,L2real,L2complex,L1complex,newDeterminant, determinant;

cout<<"請輸入三個係數";

cin>>a>>b>>c;

if ((b\*b)-(4\*a\*c)>0){

determinant=(b\*b)-(4\*a\*c);

L1=((-b)+sqrt(determinant))/(2\*a);

L2=((-b)-sqrt(determinant))/(2\*a);

cout<<"解為"<<L1<<endl<<L2<<"(二實根)";

}

if ((b\*b)-(4\*a\*c)==0){

L1=L2=(-b)/(2\*a);

cout<<"解為"<<L1<<"(重根)";

}

if ((b\*b)-(4\*a\*c)<0){

L1real=(-b)/(2\*a);

L2real=(-b)/(2\*a);

newDeterminant=(-1)\*((b\*b)-(4\*a\*c));

L1complex=(sqrt(newDeterminant)/(2\*a));

L2complex=(-1)\*(sqrt(newDeterminant)/(2\*a));

cout<<"解為"<<L1real<<L1complex<<"i"<<endl<<L2real<<L1complex<<"i"<<"(二虛根)";

}

return 0;

}

#include <iostream>

#include <cmath>

using namespace std;

/\* run this program using the console pauser or add your own getch, system("pause") or input loop \*/

int main(int argc, char\*\* argv) {

float a,b,c,L1,L2,L1real,L2real,L2complex,L1complex,newDeterminant, determinant;

cout<<"請輸入三個係數";

cin>>a>>b>>c;

if ((b\*b)-(4\*a\*c)>0&&a!=0){

determinant=(b\*b)-(4\*a\*c);

L1=((-b)+sqrt(determinant))/(2\*a);

L2=((-b)-sqrt(determinant))/(2\*a);

cout<<"解為"<<L1<<endl<<L2<<"(二實根)";

}

if ((b\*b)-(4\*a\*c)==0&&a!=0&&b!=0&&c!=0){

L1=L2=(-b)/(2\*a);

cout<<"解為"<<L1<<"(重根)";

}

if ((b\*b)-(4\*a\*c)<0&&a!=0){

L1real=(-b)/(2\*a);

L2real=(-b)/(2\*a);

newDeterminant=(-1)\*((b\*b)-(4\*a\*c));

L1complex=(sqrt(newDeterminant)/(2\*a));

L2complex=(-1)\*(sqrt(newDeterminant)/(2\*a));

cout<<"解為"<<L1real<<"+"<<L1complex<<"i"<<endl<<L2real<<"+"<<L1complex<<"i"<<"(二虛根)";

}

if (a==0&&b==0&&c==0){

cout<<"無限多組解";

}

if (a==0&&b!=0&c!=0){

cout<<"無一元二次解";

}

return 0;

}