NextDate function.

Requirement:

"Next Date" is a function consisting of three variables like: month, date and year. It returns the date of next day as output. It reads current date as input date.

The constraints are

C1: 1 ≤ month ≤ 12
C2: 1 ≤ day ≤ 31
C3: 1812 ≤ year ≤ 2018.

If any one condition out of C1, C2 or C3 fails, then this function produces an output "value of month not in the range 1...12".

Since many combinations of dates can exist, hence we can simply displays one message for this function: "Invalid Input Date".

A very common and popular problem occurs if the year is a leap year. We have taken into consideration that there are 31 days in a month. But what happens if a month has 30 days or even 29 or 28 days?

A year is called as a leap year if it is divisible by 4, unless it is a century year. Century years are leap years only if they are multiples of 400. So, 1992, 1996 and 2000 are leap years while 1900 is not a leap year.

Steps to be followed for the above lab program.

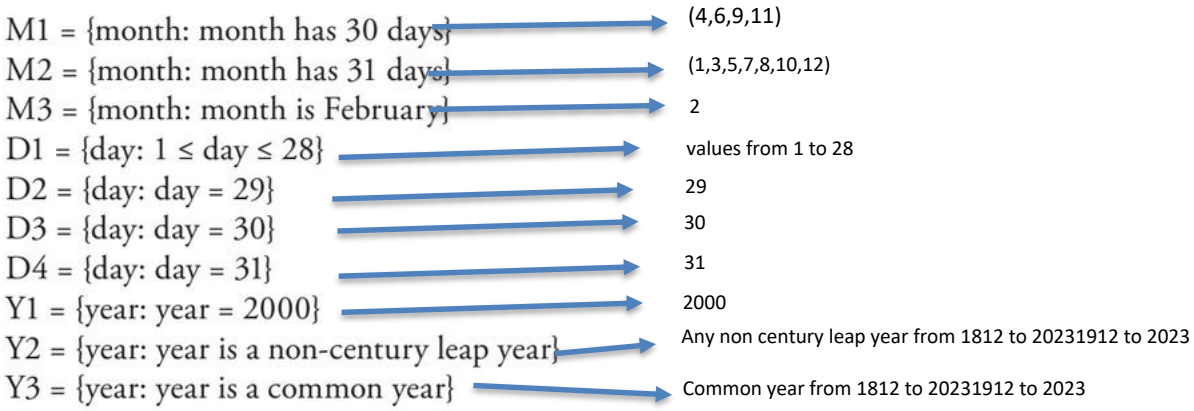
1. Write a program for the above problem statement
2. Prepare the test Input for the program
3. Write the test-cases in the lab test-case Template
4. Execute the test-cases
5. Compare the Expected output and Actual output and write the status in the test-case table.
6. Show few failed test cases by following the mutation testing.

```
#include<stdio.h>
main( )
{
Int month[12]={31,28,31,30,31,30,31,31,30,31,30,31};
int d,m,y,nd,nm,ny,ndays;
printf("enter the date,month,year");
scanf("%d%d%d",&d,&m,&y);
ndays=month[m-1];
if(y<=1812 || y>2018)
{
        printf("Invalid Input Year");
        exit(0);
}
if(d<=0 || d>ndays)
{

        printf("Invalid Input Day");
        exit(0);
}
if(m<1 || m>12)
{
        printf("Invalid Input Month");
        exit(0);
}

if(m==2)
{
        if(y%100==0)
        {
                if(y%400==0)
                ndays=29;
        }
        else
        if(y%4==0)
                ndays=29;
}
nd=d+1;
nm=m;
ny=y;
```

```
if(nd>ndays)
{
        nd=1;
        nm++;
}
if(nm>12)
{
        nm=1;
        ny++;
}
printf("\n Given date is %d:%d:%d",d,m,y);
printf("\n Next day's date is %d:%d:%d",nd,nm,ny);
}
```

Prepare the test Input for the program

Expected Output are to check **Increment in the date, Increment in the month, Increment in the year and check for the leap year**. Hence partition the input domain of date month and year as shown below.



Procedure: Suppose to pick only one input value from each class of the variable to form the test input.

D1	D2	D3	D4
28	29	30	31

M1	M2	M3
4	12	2

Y1	Y2	Y3
2000	2004	2001

Prepare Test case for

1. Weak Normal ECT
2. Strong Normal ECT
3. Weak Robust ECT
4. Strong Robust ECT

Test-case Table for Next Date Program using Equivalence Class Testing

Project Name	Next Date Program
Module Name	Checking for Tommorow's Date
Creation By	
Creation Date	
Reviewed By	
Reviewed Date	

Test Scenario ID	Test Scenario Description	Test Case ID	Test Case Description	Test Steps	Pre-conditions	Test Data	Expected Output	Actual Output	Status
TS_Tomorrows Date	Check what is tomorrow's date is	TC_ND_001	Test for Increment in the date	Enter the Input for locks stocks and barrels from C1	Enter input should be valid month and year	dd=28 mm=4 and yyyy=2000	29-04-2000		
		TC_ND_002	Test for Increment in the month	Enter the Input for DD , MM and YYYY from C1	Enter input should be valid month and year				
		TC_ND_003	Test for Increment in the year						