Program 2:

Design, develop, code and run the program in any suitable language to implement the NextDate function. Analyze it from the perspective of equivalence class value testing, derive different test cases, execute these test cases and discuss the test results.

Code:

```
#include<stdio.h>
int check(int day,int month)
{
if((month==4||month==6||month==9 ||month==11) && day==31)
return 1;
else
return 0;
}
int isleap(int year)
{
if((year%4==0 && year%100!=0) || year%400==0)
return 1;
else
return 0;

}
int main()
{
int day,month,year,tomm_day,tomm_month,tomm_year;
char flag;
do
{
flag='y';
printf("\nenter the today's date in the form of dd mm yyyy\n");
scanf("%d%d%d",&day,&month,&year);
tomm_month=month;
tomm_year= year;
if(day<1 || day>31)
{
printf("value of day, not in the range 1...31\n");
flag='n';
}
if(month<1 || month>12)
{
printf("value of month, not in the range 1....12\n");
flag='n';
}
else if(check(day,month))
{
printf("value of day, not in the range day<=30");
flag='n';
}
```

```
if(year<=1812 || year>2013)
{

printf("value of year, not in the range 1812..... 2013\n"); flag='n';
}
if(month==2)
{
if(isleap(year) && day>29)
{
printf("invalid date input for leap year");
flag='n';
}
else if(!(isleap(year))&& day>28)
{
printf("invalid date input for not a leap year");
flag='n';
}
}
}while(flag=='n');

switch (month)
{
case 1:
case 3:
case 5:
case 7:
case 8:
case 10:if(day<31)
tomm_day=day+1;
else
{
tomm_day=1;
tomm_month=month+1;
}
break;
case 4:
case 6:
case 9:
case 11: if(day<30)
tomm_day=day+1;
else
{
tomm_day=1;
tomm_month=month+1;
}
break;

case 12: if(day<31)
tomm_day=day+1;
else
{
tomm_day=1;
tomm_month=1;
if(year==2013)
{
```

```
printf("the next day is out of boundary value of year\n");  tomm_year=year+1;
}
else
tomm_year=year+1;
}
break;

if(day<28)
tomm_day=day+1;
else if(isleap(year)&& day==28)
tomm_day=day+1;
else if(day==28 || day==29)
{
tomm_day=1;
tomm_month=3;
}
break;
}
case 2:
printf("next day is : %d %d %d",tomm_day,tomm_month,tomm_year);
return 0;}
```

Test Case Name : Equivalence class test cases for Next date

Experiment Number : 12

Test data : Enter the three integer value

Pre-condition : Month 1 to 12 , DAY 1 TO 31 AND YEAR 1812 TO 2013

Valid Cases

M1 = { month ; 1 ≤ month ≤ 12 }

D1 = { day : 1 ≤ day ≤ 31 }

Y1 = { year : 1812 ≤ year ≤ 2013 }

Invalid cases

M2 = {month : month < 1}

M3 = {month : month > 12}

D2 = {day : day < 1}

D3 = {day : day > 31}

Y2 = {year : year < 1812}

Y3 = {year : year > 2013}

Next date Output Equivalence Class Testing

(Weak and Strong Normal Equivalence Class)

Case Id	Description	Input Data			Expected Output			Actual output			Status	Comment
		month	day	year	month	day	year	month	day	year		
WN1,SN1	Enter the M1, D1 and Y1 valid cases	6	15	1912	6	16	1912					

(Weak Robustness Equivalence Class)

Case Id	Description	Input Data			Expected Output			Actual output			Status	Comment
		month	day	year	month	day	year	month	day	year		
WR1	Enter the M1, D1 and Y1 cases	6	15	1912	6	16	1912					
WR2	Enter the M2 , D1 and Y1 cases	-1	15	1912	Should display the message value of the month not in the range 1..12							
WR3	Enter the M3 ,D1 and Y1 cases	13	15	1912	Should display the message value of the month not in the range 1..12							
WR4	Enter the M1, D2 and Y1 cases	6	-1	1912	Should display the message value of the day not in the range 1..31							
WR5	Enter the M1, D3 and Y1 cases	6	32	1912	Should display the message value of the day not in the range 1..31							
WR6	Enter the M1, D1 and Y2 cases	6	15	1811	Should display the message value of the year not in the range 1812..2013							
WR7	Enter the M1, D1 and Y3 cases	6	15	2014	Should display the message value of the year not in the range 1812..2013							

(Strong Robustness Equivalence Class)

Case Id	Description	Input Data			Expected Output	Actual Output	Status	Comment
		month	day	year				
SR1	Enter the M2 , D1 and Y1 cases	-1	15	1912	Should display the message value of the month not in the range 1..12			
SR2	Enter the M1, D2 and Y1 cases	6	-1	1912	Should display the message value of the day not in the range 1..31			
SR3	Enter the M1, D1 and Y2 cases	6	15	1811	Should display the message value of the year not in the range 1812..2013			
SR4	Enter the M2 , D2 and Y1 cases	-1	-1	1912	(i)Should display the message value of the month in range 1..12			
					(ii) Should display the message value of the day in range 1..31			
SR5	Enter the M1, D2 and Y2 cases	6	-1	1811	**(i)** Should display the message value of the day in range 1..31			
					(ii) Should display the message value of the year in range 1812..2013			
SR6	Enter the M2, D1 and Y2 cases	-1	15	1811	(i) Should display the message value of the month in range 1..12			
					(ii) Should display the message value of the year in range 1812..2013			
SR7	Enter the M2, D2 and Y2 cases	-1	-1	1811	(i)Should display the message value of the month in range 1..12			
					(ii) Should display the message value of the day in range 1..31			
					(iii) Should display the message value of the year in range 1812..2013			

Some addition equivalence Boundary checking

Case Id	Description	Input Data			Expected Output			Actual Output			Status	Comment
		day	month	year	day	month	year	day	month	year		
1	Enter the D1, M1 and Y1 valid cases	31	12	1811	Should display the message value of the year in range 1812..2013							
2	Enter the D1, M1 and Y2 valid cases	31	12	2012	1	1	2013					
3	Enter the D1, M1 and Y3 valid cases	31	12	2013	Should display the message Next is out of boundary 2013							

Program 3:

Design, develop, code and run the program in any suitable language to solve the commission problem. Analyze it from the perspective of decision table-based testing, derive different test cases, execute these test cases and discuss the test results.

Program:

/* Assumption price for lock=45.0, stock=30.0 and barrels=25.0 production limit could sell in a month 70 locks,80 stocks and 90 barrels commission on sales = 10 % <= 1000 and 15 % on 1000 to 1800 and 20 % on above 1800*/

Code:

```
#include<stdio.h>
int main()
{
int locks, stocks, barrels, tlocks, tstocks, tbarrels;
float lprice, sprice, bprice, sales, comm;
int c1,c2,c3,temp;
lprice=45.0;
sprice=30.0;
bprice=25.0;
tlocks=0;
tstocks=0;
tbarrels=0;
printf("\nenter the number of locks and to exit the loop enter -1 for locks\n");
scanf("%d",&locks);
while(locks!=-1)
{
c1=(locks<=0||locks>70);
printf("enter the number of stocks and barrels\n");
scanf("%d%d",&stocks,&barrels);
```