Control Flow Test Design

Consider the following Python code for the next date function

```
1. def getNextDate(year, month, day):
2.       if (year < 0):
3.         raise ValueError("Wrong value for year")
4.       if (year % 400 == 0):
5.         leap_year = True
6.       elif (year % 100 == 0):
7.         leap_year = False
8.       elif (year % 4 == 0):
9.         leap_year = True
10.      else:
11.        leap_year = False
12.      if month in (1, 3, 5, 7, 8, 10, 12):
13.        month_length = 31
14.      elif month == 2:
15.        if leap_year:
16.           month_length = 29
17.        else:
18.           month_length = 28
19.      elif month in (4, 6,9, 11):
20.           month_length = 30
21.      else:
22.           raise ValueError("Wrong value for month")
23.      if day < 1 or day > month_length:
24.           raise ValueError("Wrong value for day")
25.      elif day < month_length:
26.          day += 1
27.      else:
28.          day = 1
29.          if month == 12:
30.             month = 1
31.             year += 1
32.          else:
33.             month += 1
34.      return year, month, day
```

Question 1

Draw the simplified control flow graph corresponding to function ***getNextDate***.

Question 2

Give a minimal test suite to achieve 100% instruction coverage of function ***getNextDate***.

Question 3

Give a minimal test suite to achieve 100% branch coverage of function ***getNextDate.***

Question 4

Give a minimal test suite to achieve 100% branch/condition coverage of function `getNextDate`**.**

Question 5

Is it possible to derive a test suite to achieve 100% path coverage of function **`getNextDate`**? Justify your answer