ESC101 : Fundamental of Computing

Lab 5 for 1st September 2008

1. **Date**(marks=10)
   There is a single number representation for any date in this century : For example, 20081125 corresponds to **25 November 2008**, and 20890612 corresponds to **12 June 2089**. In this convention, the last two digits represent the day of the month, the next two digits represent the month, and the remaining number represents year. In this convention, it can be seen that 20082410 is not a valid date since 24 is not a valid month number. In a similar way, 20080231 is not a valid date since February does not have 31 days. Write a program in which you declare a variable `date` of type `int`. Assign it some positive number. If it represents a valid date based on the above convention, the program should print the date on the screen, otherwise it should print the suitable message about invalidity of the date.
   Examples :
   if date = 20081211, the output should be : **11 December 2008**.
   if date = 20080229, the output should be : **29 February 2008**.
   if date = 20890631, the output should be : **June can't have 31 days**.
   if date = 20101331, the output should be : **13 is invalid month**.

   Hint : You may find the use of `switch` statement useful here. Remember that a leap year is divisible by 4 and has 29 days in February.

2. **Precision upto any number of places**(marks=10)
   In the class we discussed that there is only finite precision supported by float and double. However, we may achieve arbitrarily high degree of precision if we program carefully. This toy problem is to manifest this fact.

   Declare two variables `a` and `b` of type int. Assign positive values to `a` as well `b` such that `a<b`. Write a program which prints the **correct** value of `a/b` upto 60 decimal places. For example,

   for `a`=1, `b`=8, the output should be
   0.125

   for `a`=1, `b`=1231, the output should be

   0.000812347684809098294069861900893582453290008123476848090982