ESC101 : Fundamental of Computing

Lab 5 for 2nd September 2008

1. **Remainder of a large number when divided by a small number :** (marks=10)
   Declare three variable `a` , `b` and `c` of type `int`. Write a Java program to find the remainder when a^b is divided by c. For example, when $a = 3$, $b = 247$ and $c = 17$ the output should be 11, that is, the remainder when we divide 3^{247} by 17 is 11. Your program should work on any input which satisfies $a < c$ and $2 \leq c \leq 200$.
   Hint: $(x \times y)\%z = ((x\%z) \times (y\%z))\%z$.

2. **Depth of function :** (marks=10)
   There is a function f defined for all positive integers such that $f(n)$ is equal to the sum of square of digits of n in decimal representation. For example,
   $f(14) = 17$
   $f(192) = 86$
   $f(1025) = 30$
   It is a mathematical fact that starting from any arbitrary positive integer n, if we keep on applying f, we shall eventually get either 1 or 89. For example,
   $f(f(f(f(f(f(f(45)))))))) = 89$.
   $f(100) = 1$.
   $f(f(f(f(f(f(f(f(89)))))))) = 89$.

   For each number n, the smallest non zero number of times we need to apply function f till we get either 1 or 89 is called $depth\_f(n)$. From the above examples, it follows that $depth\_f(45) = 7$, $depth\_f(89) = 8$, $depth\_f(100) = 1$. Write a program to compute the integer n in the range $[2, 99]$ which has maximum $depth\_f$ value.

Note : one mark will be deducted from each question if the code is not properly indented. The students are encouraged to write comments to improve the readability of the code.