ESC101 : Fundamental of Computing
Mock Lab Test for 15th September 2008

Instructions:

1. The duration of the test is **2:00 pm to 5:00 pm**.
2. **Directory Structure:** Create a directory inside your home directory. Name it with your roll number. e.g. If your roll number is $Y8001$, the directory should be named $Y8001$. Create two files inside this directory: $PerfectNumber.java$ and $CircularPrime.java$. Write the solution to the first problem in $PerfectNumber.java$ and the solution to the second problem in $CircularPrime.java$.
3. Please use *meaningful* identifiers for variables and methods. Use comments to improve readability of the program. Properly indent your code. Otherwise some marks may get deducted irrespective of whether your program is correct.

Problems:

1. **Perfect, Abundant or Deficient number :** (marks=10)
   The *proper divisors* of an integer n are defined as the positive divisors of n other than n itself.

   e.g. the proper divisors of 10 are 1, 2 and 5.

   A number is called a *Perfect Number* if the summation of all of its proper divisors is *equal* to the number itself. For example, 6 and 28 are perfect numbers because

   $1 + 2 + 3 = 6$ and,
   $1 + 2 + 4 + 7 + 14 = 28$.

   A number is called an *Abundant Number* if the summation of all of its proper divisors is *greater* than the number. For example, 12 is an abundant number because the sum of its divisors,

   $1 + 2 + 3 + 4 + 6 = 16 > 12$

   A number is called a *Deficient Number* if the summation of all of its proper divisors is *less* than the number. For example, 9 is a deficient number because the sum of its divisors,

   $1 + 3 = 4 < 9$

   Write a JAVA program that takes a positive integer n as command line argument and prints whether it is "Perfect", "Abundant" or "Deficient".

2. **Checking if two numbers are permutations of each other :** (marks=10)
   Write a JAVA program that takes two positive integers $n1$ and $n2$ from the command line and checks if they are permutation of each other.