CISC106 Summer 2011 Lab04

- This lab an all subsequent labs will be due Sunday at 11:55 PM EDT on Sakai.
- a lab04_tests.py skeleton is provided for you on the course website.

Preparation (do not submit for grading)

1. Run Python in interactive mode (by simply giving the `python-2.7` command without any arguments.) At the prompt, enter the following two commands:

```
>>> import Tkinter
>>> Tkinter._test()
```

   A small window should pop up with two test buttons. Clicking the 'Click me' button will add brackets to its description. Clicking quit will close the window. If the window does not appear, ask the TA for help.

Programs (to be graded)

1. Download lab04.py from the course website. Run it and try entering some numbers for Rate and Data. Notice how it always tells you your bill will be $0, no matter what you enter (so long as you enter valid values.)
2. Now open lab04.py in your editor. Note that there is quite a bit of code here! Fortunately, we don't care about most of it and as such you can ignore it. Look for a function that seems to have something to do with calculating a bill - you'll need to give this function an implementation based on the following critera:

   "The internet service provider charges a base rate per megabyte (MB) transferred depending on market conditions. In addition to the base, transfers between 100 and 500 MB are charged an additional $0.05 per MB plus 33% of the base. Data transfers between 500 MB and 1500 MB are charged 1.44 times the base plus $0.08 per MB. Above 1500 MB the rate is simply twice the base. Transfers are 'tiered', *e.g.* a transfer of 400 MB would be charged the base rate for the first 100 MB and 133% of the base + $0.05 per meg for the other 300 MB."

   After you're done, run the program again. The calculator should be giving correct values now. Modifying existing code is a common task in software engineering - often times this code will be part of a large system which wasn't written by you. Making small, calculated changes to existing code is also the best way to learn that codebase.

   Now that the calculator is working, write a unit test for your function. You should check five different cases in your test (think about what these should be.)

You should submit lab04.py and lab04_tests.py along with any other docs required by your TA on Sakai.