Department of Computer Science & Engineering
IIT Bombay
CS 699 Software Lab Mid-semester Examination
Practical Problem Solving in Limited Time
Saturday Sept. 11, 2016, 9:30-12:30

General Instructions: If one version of your program works on a test case, but has bugs for another, don't destroy it. Else you lose your good versions and you will end up submitting a version that works for no test case! This happened for some students last time.
You can Download your own moodle submissions and Resources from Moodle before the exam starts. Do not use network when the examination starts unless we specify it if necessary. Use of network during exam will result in disqualification for this exam. Your IP address will be noted down on attendance sheet. The network can be used once again to upload your submissions at the end of examination. All the logs are available for analysis later. So do not even attempt to communicate with others through network, just concentrate on what you can do. Enjoy Programming, All the best!

1. **A Simple class diagram generation tool as follows:**
   Input format: as given in test file test.u
   Output we need: picture pdf as given in test.pdf

   How to do it: Write an Awk script to convert the .u file to a .dot file Then use Graphviz (as in command 'dot -Tpdf xyz.dot > xyz.pdf') to produce the picture pdf.

   What to submit: (1) your awk script (2) pdf that your program generates for test.u (3) One more test case .u file of your own, and it's pdf generated using your awk program. Add these into folder named *Problem1*.

   How we will evaluate it: We will run your awk script and convert your .u files to .dot files by your awk script. Then we will generate the final pdf from .dot by using graphviz. We will also use our hidden test cases to test your program. We will use similar scheme for all problems.

   Observe that the input format is organized as follows: There are two tags *classname:* and *pubmember:* Under each classname there can be any pubmember statements. But one pubmember per line. The values

of classname and pubmember are to be taken as whatever appears till the end of that line.

2. Given a marks datafile "testdata1", write an Awk script to grade the students with the following marking scheme. The grading scheme is as follows: 80-100(AA), 60-79(AB), 40-59(BB), 25-39(BC), and (<25) RETAG. The grade should be added to another column. The output is to be produced in another file called "outdata1". Sample files are given.

   What to submit: (1) your awk script (2) output that your program generates for "testdata1" (3) One more test case "testdata2" file of your own, and it's output generated using your awk program in "testdata2". Add these into folder named *Problem2*.

3. Write a gnuplot script which will use the output of the above program to generate (1) plot1.png which shows how many students got a particular grade using a bar chart (2) plot2.png which shows marks vs. grades plot, with grades on y axis, and marks sorted on x axis.

   What to submit: (1) your gnuplot script (2) plots for your "outdata-data1" (3) Plots for your "outdata2". Add these into folder named *Problem3*.

4. Write a latex file to produce a part of this test paper with all its styles, fonts, enumerations and spaces. The text matter is given to you as file "text". Use it and make it into valid latex file manually called "paper.tex". Produce pdf and call it "paper.pdf". Submit both files in folder "Problem2"