To print higher-resolution math symbols, click the
Hi-Res Fonts for Printing button on the jsMath control panel.

Lab 3 Grading Guidelines last modified by Aaron Bloomfield on September 19, 2013 11:28:46 AM EDT

Lab 3 grading guidelines

Execution runs

The execution runs for this lab (for the in-lab and post-lab) were the same as those on the board during the lab day, and all evaluate to 216.

- 8 9 - 5 * -3 ~ - 4 * 9 * ~ 8 9 * -
- 6 4 + 5 / 3 * 6 6 * *
- 3 4 + 7 / 4 - ~ 3 * 2 2 2 * * * 3 *
- 220 4 -
- 1296 6 /

Pre-lab

Out of 10 points. This lab part **is** expected to compile.

Point Assignment

- 2 points: Asked an intelligent question in unix.questions.txt (if it's an obviously unintelligent question, give 1 point here)
- 8 points: The pre-lab code works. Note that they will have hard coded their answer, so it will produce the same amount each time. We want to check that they (generally) test all the operators
  - 8 points: if it works properly for all the operators
  - 6 points: if it mostly works, but one or two operators are not working properly
  - 4 points: they had the right idea, but it doesn't work very well
  - 2 points: they did some work, but didn't get it working
  - 0 points: nothing was submitted

Feel free to interpolate between these values.

In-lab

Out of 10 points. This lab part **is** expected to compile.

This grading scheme is similar to that of the pre-lab, but on a different scale (as there was no unix.questions.txt to be checked). Furthermore, a calculator that is not working resulted in a somewhat more points deducted (percentage-wise), as they had more time to get it working.

Point Assignment

- 10 points: The in-lab code works
  - 10 points: if it works properly for all the test cases
  - 7 points: if it mostly works, but one or two operators are not working properly
  - 4 points: they had the right idea, but it doesn't work very well
  - 2 points: they did some work, but didn't get it working
  - 0 points: nothing was submitted

Feel free to interpolate between these values.

If their code had problems with reading in input from a pipe, but works fine via keyboard input entry, they can get full credit. Likewise, if it *only* works via a pipe, then they still can get full credit.

Post-lab

Out of 10 points. This lab part **is** expected to compile.

Point Assignment

- 5 points: Proper code execution
  - Their code should correctly execute for each of the five input runs. 1 points for each input run that results in 216.
  - If their code had a run-time error because it required additional input, they still get full credit (assuming it works properly, of course)
- 4 points: Quality of stack code
  - 4 points: Stack code looks like it should work in theory or does work completely.
  - 3 points: If more than half of the stack code looks unintelligible or student used an array to store the data rather then nodes as it creates scalability issues.
  - 1 points: Almost no work done.
  - 0 points: No work done.
- 1 point: difficulties.txt. Award 1.0, 0.5, or 0.0 points for their paragraph, depending on if it was good, bad, or absent (respectively, of course). We aren't looking for a essay here, but a decent description of the problems they encountered.
- 7 points were taken off if the STL stack implementation was still being used.

Feel free to interpolate between these values.

If their code had problems with reading in input from a pipe, but works fine via keyboard input entry, they can get full credit. Likewise, if it *only* works via a pipe, then they still can get full credit.

Untitled note

Find a notebook

Add tag

Add comments

Version 6.0.6: c128369/1.0.1.250

Web Clipper tutorial

Options

Saving clip...

To:

Share

Save

Selection

PDF