Programming Languages G22.2110 Spring 2007 hw04

Assigned Tu 2/6/2007, due We 2/14/2007 at 1pm.

How to Submit Homework Assignments

Email your answers, in either plain text format or as pdf, to Long Lin <llin@cs.nyu.edu>. Assignments are due on Wednesdays at 1pm. This deadline will be strictly enforced.

Reading Assignments

• For lecture on 2/6/2007: Scott 3.1-3.3
• For lecture on 2/13/2007: Scott 7.7.1-7.7.2 and Drexel 1-10 (http://einstein.drexel.edu/courses/CompPhys/General/C basics/)

Homework Assignments

1. Practice your Scheme skills by writing some more code. Specifically:

   1a. (5 points) Write a recursive function list-sum that takes one argument, a list $x$, and returns the sum $\sum_{i=1}^{n} x_i$ of the numbers in the list. For example:

   (list-sum '(1 2)) ⇒ 3
   (list-sum '(2.4 -2.1 5 4.23)) ⇒ 9.53

   1b. (5 points) Write a function arithmetic-mean that takes one parameter, a list $x$, and returns the arithmetic mean $\bar{x} = \frac{1}{n} \sum_{i=1}^{n} x_i$ of the numbers in the list. For example:

   (arithmetic-mean '(1 2)) ⇒ 1.5
   (arithmetic-mean '(2.4 -2.1 5 4.23)) ⇒ 2.3825

   1c. (5 points) Write a recursive function standard-deviation-helper that takes two parameters, a list $x$ and a number $\bar{x}$, and returns the result of $\sum_{i=1}^{n} (x_i - \bar{x})^2$ where the $x_i$ are the numbers in the list. For example:

   (standard-deviation-helper '(1 2) 1.5) ⇒ 0.5
   (standard-deviation-helper '(2.4 -2.1 5 4.23) 2.3825) ⇒ 30.357675

   1d. (5 points) Write a function standard-deviation that takes one parameter, a list $x$, and returns the standard deviation $\sigma = \sqrt{\frac{1}{n} \sum_{i=1}^{n} (x_i - \bar{x})^2}$ of the numbers in the list. For example:

   (standard-deviation '(1 2)) ⇒ 0.5
   (standard-deviation '(2.4 -2.1 5 4.23)) ⇒ 2.75488997057959
2. Answer Questions 13, 15, and 18 from Page 124 of the Scott book. (13: 3 points, 15: 3 points, 18: 3 points)

3. Answer Question 3.5 from Page 152 of the Scott book (a: 5 points, b: 8 points, c: 6 points). Your answer to part (b) should start with the table below, and add more activation records on top of the activation record for “main” shown in the table.

<table>
<thead>
<tr>
<th>addr</th>
<th>routine</th>
<th>variable</th>
<th>value</th>
</tr>
</thead>
<tbody>
<tr>
<td>97</td>
<td>main</td>
<td>g</td>
<td>(fill in value of g)</td>
</tr>
<tr>
<td>98</td>
<td>static</td>
<td>link</td>
<td>-</td>
</tr>
<tr>
<td>99</td>
<td>dynamic</td>
<td>link</td>
<td>-</td>
</tr>
</tbody>
</table>

4. Start teaching yourself C by doing the following. (If you already know C, you can skip Step 4a, but please do Steps 4b-4d, and please offer your help to anyone who needs a “guru” for Step 4a).

4a. If possible, find peers (other students who want to learn C together with you) and gurus (people who already know C, whom you can ask questions when you get stuck).

4b. Make sure you have access to gcc (the GNU C compiler), and that you can successfully compile and run programs. It should be installed on any Linux or Unix machine. Alternatively, you can install it on Windows with Cygwin, or on Mac OS X with the developer CD.

4c. Read the first 10 sections of the C tutorial (see reading assignment above). Along the way, you should try things out with gcc.

4d. (2 points) What is the URL of a reference for the C standard library? Unfortunately, the C language definition is not available online; if you are unfamiliar with C, I recommend you buy and read “The C Programming Language” (2nd edition) by Kernighan and Ritchie.

http://www.cs.nyu.edu/courses/spring07/G22.2110-001/hw04.pdf
Total points: 50.