

COMP 210, Spring 2000, Homework 8
Due Friday, April 21, 2000 at the start of class

Before you start the homework, you should remind yourself of our General Advice, Advice on Homeworks, and Grading Guidelines. All are available from the class web site (<http://www.owl.net.rice.edu/~comp210>).

Running Average

1. (2 pts) Develop a program, `avg`, that takes as input a number and outputs the average of all numbers passed to `avg` since it was created. Make sure that any information used by `avg` to record its past history is hidden.

Owlnet Accounting

2. (2 pts) Owlnet has a problem. It needs robust software so that it can charge students for each page that they print. Since the software must be flawless (after all, they lose money if it doesn't work), they would like the COMP 210 class to develop it for them. For each student, they must record a student-identification number and the number of pages printed in the current month. You should design the data structures needed to implement the following programs:
 - a) (2 pts) `add-account: number → boolean`
This program takes a student-identification number and creates an account in your database of accounts. It should initialize the number of pages printed to zero. It returns true if the account is added successfully, and false if the account already exists.
 - b) (2 pts) `charge-account: number number → boolean`
This program takes a student-identification number and a number of pages printed and updates the charging record for the student to reflect the newly printed pages. It returns true if the student's record is found, and false otherwise.
 - c) (2 pts) `report-accounts: void → void`
This program traverses the entire database and prints out, for each student-identification number, the number of pages printed during the month. The report should be printed with one account per line. It should print in ascending order by student-identification number.
[Hint: add-account can insert new accounts in the appropriate order.]
 - d) (2 pts) `clear-accounts: void → void`
This program traverses the entire database and resets every student's count of printed pages to zero. It is used right after `report-accounts` to avoid double charging the students.

[Hint: Tim always says that if you're going to touch every element of the list, you should use an abstract function.]