

**Concordia University
Department of Computer Science
and Software Engineering**

**Introduction to Programming
COMP 248 --- Winter 2006 --- Section U**

Assignment #2

Deadline:	week 6, Friday February 10 th , 2006
Evaluation:	3% of final mark
Late submission:	not accepted
Teams:	this is an individual assignment
Purpose:	to help you practice the material presented in Chapter 2

Exercises

1. The Babylonian algorithm to compute the square root of a number n is as follows:

1. *Make a guess at the answer (you can pick $n/2$ as your initial guess)*
2. *compute $r = n/guess$*
3. *set $guess = (guess+r)/2$*
4. *go back to step 2 for as many iterations as necessary. The more you repeat step 2 and 3, the closer guess will become to the square root of n .*

Write a program that inputs an integer for n , iterates through the Babylonian algorithm five times (without using a loop construct), and outputs the answer as a `double` to two decimal places.

2. Write a program that reads in a line of text and then two words (separately), and then outputs that line of text with the first occurrence of the first word replaced with the second word. For example, a possible sample dialog might be:

```
Enter a line of text:
I hate you.
Enter a word to be replaced in this line:
you
Enter a word to replace this one by:
it
I have rephrased that line to read:
I hate it.
```

You can assume that the first word occurs in the input. If the first word occurs more than once in the input line, your program should replace only the first occurrence of it.

3. In US units, a metric ton is 35,273.92 ounces. Write a program that will read the weight of a package of breakfast cereal in ounces and output the weight in metric tons as well as the number of boxes needed to yield one metric ton of cereal.

Evaluation Criteria (for each individual exercise)

Comments: 6 pts	
Description of the program (authors, date, purpose):	2 pts
Description of variables, constants and algorithm:	4 pts
Programming style: 8 pts	
Use of constants where necessary:	2 pts
Use of significant names for identifiers:	2 pts
Indentation and readability:	2 pts
Simplicity of the algorithm:	2 pts
Results on test data and on unseen data:	<u>6 pts</u>
Total	20 pts

Submission Procedure

When you are finished all programs, you should submit a paper version and an electronic version of your java source files. **Both** the paper **and** the electronic submission must be on time; otherwise, the assignment will be considered late.

1. Paper submission:

- Fill out an expectation of originality form (given to you in class or available on the Web page).
- Print all your source code (the `.java` files).
- Print a screen shot of your output with the data given above.
- Staple everything together and give it to your instructor in class. If you cannot give the assignment in class, go to the Department of Computer Science and Software Engineering secretary desk in EV3.139, ask the receptionist to write the date and time on your assignment and initial it, and put the assignment in your instructor's assignment submission box, located in EV3.414.

2. Electronic submission:

Create one zip file, containing all source files for your assignment. The zip file should be called `assignment#_studentID`, where # is the number of the assignment and `studentID` is your student ID number. For example, for the first assignment, student 123456 would submit: `assignment1_123456.zip`. Use the Electronic Submission Form to upload your zip file. Go to the following URL, fill-in the form, and submit your file: <https://eas.encs.concordia.ca/eas/authentication.jsp>,