

Assignment	Points	Announced	Due
#2	40	9/10	9/21

Calculators

1 Overview

In this lab, you will work on two different problems: a time calculator and a tax calculator.

2 Learning Outcomes

By the end of this project students should be able to:

- write, save, and evaluate simple programs;
- read and write programs with numerical literals;
- read and write programs with simple function calls (e.g., input, print);
- read and write programs with conditionals;
- break up simple problems into multiple steps;
- work effectively with a partner using pair-programming;
- write an effective report that describes the students' problem-solving process.

3 Pre-Lab Instructions

Do this part before you come to lab:

- Review chapters 4 and 5 from the textbook and the in-class exercises: checkerboard and Roman numerals. Attempt solving these two exercises by yourself.
- Read this blog post about comparing strings in Java:
<https://alvinalexander.com/java/edu/qanda/pjqa00001.shtml>
- Read this Wikipedia article about magic numbers:
[https://en.wikipedia.org/wiki/Magic_number_\(programming\)](https://en.wikipedia.org/wiki/Magic_number_(programming))

4 Submission Instructions

1. Use a basic text editor (e.g. Notepad, Notepad++, Atom...) and **not an IDE** for this lab assignment.
2. In addition to the lab report, submit two Java files named `TaxCalculator.java` and `TimeCalculator.java`
3. Don't submit .class files nor .jar files nor archives (e.g. .zip, .rar ...)

5 Lab Instructions

Do this part in lab:

1. The tax calculator problem

You are required to write an income tax calculator program (`TaxCalculator.java`). Your program must ask the user for input that is sufficient to calculate the owed tax. Use the following schedule:

If your status is Single and if the taxable income is over	but not over	the tax is	of the amount over
\$0	\$8,000	10%	\$0
\$8,000	\$32,000	\$800 + 15%	\$8,000
\$32,000		\$4,400 + 25%	\$32,000
If your status is Married and if the taxable income is over	but not over	the tax is	of the amount over
\$0	\$16,000	10%	\$0
\$16,000	\$64,000	\$1,600 + 15%	\$16,000
\$64,000		\$8,800 + 25%	\$64,000

Begin by using a sheet of paper to design this program with your partner. What data do you need to get from the user? How will you get the data from the user? What variable types will you use? How many if statements are needed? Then, create a file called “TaxCalculator.java” and convert your design to Java code. Below is a sample screenshot.

```

Enter your status (Single or Married):Single
Enter your taxable income: 20000
You owe $2600.00 in tax.

Enter your status (Single or Married):Single
Enter your taxable income:50000
You owe $8900.00 in tax.

Enter your status (Single or Married):Married
Enter your taxable income:20000
You owe $2200.00 in tax.

Enter your status (Single or Married):Married
Enter your taxable income:100000
You owe $17800.00 in tax.

```

2. The time calculator problem

Write a Java program (TimeCalculator.java) that asks the user to enter a number of seconds. There are 60 seconds in a minute. If the number of seconds entered by the user is greater than or equal to 60, the program should display the number of minutes in that many seconds. There are 3600 seconds in an hour. If the number of seconds entered by the user is greater than or equal 3600, the program should display the number of hours in that many seconds. There are 86400 seconds in a day. If the number of seconds entered by the user is greater than or equal to 86400, the program should display the number of days in that many seconds. Use final variables for all magic numbers. Below is a sample screenshot.

Enter number of seconds:100
100 seconds is equal to 1 minute(s) and 40 second(s).

Enter number of seconds:120
120 seconds is equal to 2 minute(s) and 0 second(s).

Enter number of seconds:4000
4000 seconds is equal to 1 hour(s), 6 minute(s) and 40 second(s).

Enter number of seconds:4500
4500 seconds is equal to 1 hour(s), 15 minute(s) and 0 second(s).

Enter number of seconds:100000
100000 seconds is equal to 1 day(s), 3 hour(s), 46 minute(s), and 40 second(s).

Enter number of seconds: 250000
250000 seconds is equal to 2 day(s), 21 hour(s), 26 minute(s), and 40 second(s).

6 Lab Report

Each pair of students will write a single lab report together and each student will turn in that same lab report on BBLearn. Submissions from each student on a pair should be identical.

Your lab report should begin with a preamble that contains:

- The lab assignment number and name
- Your name(s)
- The date
- The lab section number

It should then be followed by four numbered sections:

1. Problem Statement

In this section you should describe the problem in *your* own words. The problem statement should answer questions like:

- What are the important features of the problem?
- What are the problem requirements?

This section should also include a reasonably complete list of requirements in the assignment. Following your description of the problem, include a bulleted list of specific features to implement. If there are any specific functions, classes or numeric requirements given to you, they should be represented in this bulleted list.

2. Planning

In the second section you should describe what planning you did in order to solve the problem. You should include planning artifacts like sketches, diagrams, or pseudocode you may have used.

You should also describe your planning process. List the specific data structures or techniques you plan on using, and why.

3. Implementation and Testing

In the third section you should describe how you implemented your plan. As directed by the lab instructor you should (as appropriate) include:

- a copy of your source code (Submitted in BBLearn as .java files)
- a screen shot of your running application / solution
- results from testing

4. Reflection

In the last section you should reflect on the project. Consider different things you could have done to make your solution better. This might include code organization improvements, design improvements, etc.

You should also ask yourself what were the key insights or features of your solution? Were there alternative approaches or techniques you could have employed? How would these alternatives have impacted a different solution?

5. Partner Rating

Every assignment you are required to rate your partner with a score -1, 0 or +1. This should be submitted in the comment section of the BBLearn submission, and not in the report document. If you don't want to give your partner a negative rating making sure not to use a dash before listing the number! You do not have to tell your partner the rating you assign them. A rating of 1 indicates that your partner was particularly helpful or contributed exceptional effort. A rating of 0 indicates that your partner met the class expectations of them. Rating your partner at -1 means that they refused to contribute to the project, failed to put in a reasonable effort or actively blocked you from participating. If a student receives three ratings of -1 they must attend a mandatory meeting with the instructor to discuss the situation, and receiving additional -1 ratings beyond that, the student risks losing a letter grade, or even failing the course.

7 Grading Rubric

This lab assignment will be graded according to this rubric:

Criteria / Component	Points
Report	10 pts
Tax Calculator: Correct calculation of taxes for a single filer	6 pts
Tax Calculator: Correct calculation of taxes for a married filer	6 pts
Time Calculator: Correct calculation of minutes	6 pts
Time Calculator: Correct calculation of hours	6 pts
Time Calculator: Correct calculation of days	6 pts
Time Calculator: Not using final variables for magic numbers	-3 pts

Lab assignment penalties:

CS136L: Computer Science II Lab – Fall 2018

Item	Points
Missing names	-4
Missing partner rating	-2
Missing screenshots	-2
Insufficient / No commenting	-2 / -4
Too late to pair (if attended)	-4
Absent	-12
Non-compiling program	-40

Note:

If your partner is not responding to your emails and/or not collaborating, don't hesitate to reach out to the lab TA aide and/or the primary instructor.