Nassau Community College **Summer I 2016**
CSC 120 CA: Computer Science I **Prof. Christopher R. Merlo**

Project #1: Student Data
Out: Thursday, 26 May 2016
Due: Tuesday, 31 May 2016 11:59:00 pm

Please help NCC conserve resources by not printing this document on campus.

Please note that you are meant to work on this assignment yourself.

1 Objectives

- Understand program logic
- Trace code that contains assignment statements, `String` class method calls, user input statements, and output statements
- Gain experience using Eclipse
- Practice debugging a program

2 Overview

This program has been designed to help you gain valuable experience in two skills that are important for programmers to master: Tracing code and fixing logical mistakes. First, you will determine what values are stored in memory during the execution of a program. Then, you will determine how the code of a program should be rearranged so that it generates the desired output. Finally, you will take a short quiz in class on the due date to reinforce the important concepts.

3 Specifications

3.1 Understanding Program Logic

In Part 1, you must trace the program code you see below, to determine what it does. To trace the program, you must keep track of every time the value in a variable changes, and report these changes in order. You must also make note of any screen output that happens. For instance, if the variable `x` is made to store the number 15, and then the variable `y` gets the value 12, and then the phrase "y stores 12 now" is displayed, and then `x` changes again to store 21, then your report should be formatted like:

```
x = 15
y = 12
"y stores 12 now"
x = 21
```

Trace this program, assuming that the user has typed in:
Mr. Odell Beckham Jr.
and create a file called `trace.txt` with your results.

```
String fullName;
String someString;
String anotherString;
int posOfPeriod;
int numChars;
int posOfSpace;

fullName = JOptionPane.showInputDialog( "Please enter the full name" );
posOfPeriod = fullName.indexOf( "." );
System.out.println( "Position of period " + posOfPeriod );

someString = fullName.substring( 1, posOfPeriod + 5 );
System.out.println( someString );

numChars = fullName.length();
anotherString = fullName.substring( posOfPeriod + 2, numChars - 2 );
System.out.println( anotherString );

posOfSpace = anotherString.indexOf( " " );
System.out.println( "Position of space " + posOfSpace );
System.out.println( anotherString.substring( posOfSpace - 1,
                                anotherString.indexOf( "am" ) ) );
```

3.2 Debugging a Java Program

Create a new project in Eclipse called `proj1`, and create a new class within it called `Project1`. Delete the contents of the `Project1` class.

Now download the file `Project1.java` from the Project 1 website. Open the file you downloaded, copy its contents, and paste them into the new class you created.

This program is supposed to ask the user to enter his or her first and last name in a single dialog box, and then a midterm grade and a final exam grade. After all the input has been collected, the program should generate output like what you see in Figure 1 on page 3.

But if you run the program and type in **Eli Manning**, **89.5**, and **95.0**, you don't get that output. Your goal is to fix the code you downloaded so that it does generate that output. You must also write a comment along with each change you make, explaining what you changed and why.

Once the program generates the kind of output specified in Figure 1, test it again with other inputs, and make sure it works in all cases. Then, add comments at the top similar to the comments at the top of the code in Lab Assignment 1, with an appropriate title and description, and your name next to the author tag.

```
Name: Manning, Eli

There are 3 letters in my first name
There are 7 letters in my last name
There are 10 letters in my full name

First initial: E
Last initial: M

Midterm: 89.5
Final Exam: 95.0

Average: 92.25
```

Figure 1: Desired Output from Part 2

You will be submitting your edited code.

4 Analysis

In addition to your tracing log and your `Project1` source code, you must submit a plain text file called `analysis.txt` that contains your answers to the following questions:

- How much time did you spend working on this project?
- Who, if anyone, helped you with this project?
- What was the most difficult part of this project?
- Is there anything about your submission that you're unhappy with?

Your responses to these questions will be graded based upon grammar and spelling as well as on content. Please use complete sentences, proper punctuation, etc.

5 Submitting Your Project

You must submit your project using the interface on my web site:

http://www.matcmp.ncc.edu/~cmerlo/

No other submissions will be accepted. You may re-submit this project as often as you want; I will only ever see the most recent submission. Notice this means that if you submit before the deadline, and then re-submit after the deadline, your project will be late, and you will lose lateness points. You must submit the following files:

- `Project1.java`
- `trace.txt`
- `analysis.txt`

6 Assessment

It is clearly important that your program runs successfully. That is why **30%** of your grade will be based on the **correctness** of your program.

However, it is also important to write code that conforms to the rules we programmers have imposed upon ourselves, to ensure readability. **Ten percent** of your grade will be based, therefore, on your adherence to the **style guidelines** listed on my web site.

Additionally, **proper testing and proper analysis** are the tools we use to convince non-programmers that our code works. Therefore, **ten percent** of your grade will be based on your analysis.

Finally, you will take a short quiz at the end of class on the day of the deadline. The answers to this quiz will account for **50%** of your grade. You will be allowed to refer to a printout of your code during the quiz. This printout ***must*** be of your submission for `Project1.java`, and it ***must*** contain your name in the comments. It is ***strongly*** suggested that this printout include line numbers. You ***will*** receive a zero on the quiz (and therefore a maximum grade of 50 for the project) if you do not also submit a printout of your source code.

7 Things You Should Know

- It would be silly for any of us to assume that you can finish this project without any help. You're in college to be challenged. The only way you will get better is if you try to complete tasks that are outside your current level of expertise. Since this project is designed to be such a task, it is expected that you will need some assistance. **Ask for help as soon as you need it**, because if you wait, you may be too far behind to catch up. Remember that you can e-mail or IM me, or you can visit the people who work in the B 225 lab. If you e-mail me, make sure to do so using the link on my web site; otherwise, there may be a considerable delay before I reply.

- You may help each other, also, but **you are not permitted to share code.** Not even one line. Remember: unlike the lab assignments, ***you must write the code for this project yourself.*** If you're going to talk about this project with each other, leave the conversation empty-handed, and then write the code on your own. Submitting another student's code, or allowing another student to submit yours, will automatically earn both of you a zero on this assignment. Remember that earning a zero due to academic dishonesty also disqualifies you from withdrawing from the class.

- This project is due at 11:59:00 pm on May 31st. Late projects lose 10 points per 24 hour time period or portion thereof, starting at 11:59:01 on May 31st, regardless of weekends, holidays, weather, computer malfunction, etc. Any submissions uploaded after 11:59:00 pm on Friday, June 3rd will be ignored, and a grade of 0 will be recorded. (Note that the project quiz ***must*** be taken on May 31st, except in extreme cases.)

- Store your data in multiple places. Consider using a system like Dropbox (https://db.tt/KEV2lqS) or SpiderOak (http://www.spideroak.com/). (If you decide to use Dropbox, let me invite you; we both earn an extra 0.25 GB of free space.)