CMPSC 111
Introduction to Computer Science I
Fall 2013
Bob Roos
http://cs.allegheny.edu/~rroos/cs111f2013

Recitation 3
12–13 September 2013
Due in Sakai by midnight
"Checkmark" grade

Summary

Today you will get practice in writing interactive programs that can read words typed by the user using the "next()" method of the Scanner class. (In this week's lab, you will learn how to read numeric data.)

I will also ask you to fill out a short survey about the course so far; this will become a regular feature of the class.

Review the Textbook

Review the material in chapter 2, particularly the material in 2.6; this week's recitation introduces you to that material.

Warm-Up Exercise: Try This! (Not to be handed in)

The example below shows you how to have your program accept input in the form of String data typed by the user. For today, we limit ourselves to single words (no blanks allowed). Open a new file named Recit3.java and type in the program below. Pay close attention to the lines highlighted in red bold-face type:

```
import java.util.Date;
import java.util.Scanner;

public class Recit3
{
    public static void main(String[] args)
    {
        // Variable dictionary:
        Scanner scan = new Scanner(System.in); // for user input
        String lastName; // input value--user's last name
        String measure;   // input value--a unit of measure
        String veg;       // input value--a vegetable
        // Be sure to add a line to print your name and the date!

        System.out.print("Enter your last name: ");
        lastName = scan.next();

        System.out.print("Enter a unit of measurement: ");
        measure = scan.next();

        System.out.print("Enter the name of a vegetable: ");
        veg = scan.next();

        System.out.print("A tongue twister: ");
        System.out.println("Peter " + lastName + " picked a " + measure
        + " of pickled " + veg + "s.");
    }
}
```

Compile and execute the program. Note that you may enter only *single* words with no spaces. For instance, a vegetable like "`brussels sprout`" is not allowed since that's two words.

Exercise—To Be Uploaded to Sakai

Imitate the previous program, but make up your own "silly sentence." The program should ask the user for words (for instance, "Enter a noun" or "Enter the name of a city") and then assemble these into a sentence. Test it out on others in the class!

When you are finished, or at the end of the period, upload your program to your Sakai Drop Box. You do *not* have to hand in a hard copy of anything—just upload your Java program, working or not.

Survey—Required

I have emailed you a survey (look in your Allegheny email for a message labeled "Weekly Self Evaluation"). Please fill it out to the best of your ability and submit it.

General Guidelines for Recitation Sessions

- **Experiment!** Recitation sessions are for learning by doing without the pressure of grades or "right/wrong" answers. So try things! The best way to learn is by trying things out.
- **Submit *something*.** Your grade is just 0 or 1, depending on whether or not you attempt the work and upload something to Sakai.
- **Try to Finish During Class.** Recitation exercises are not intended to be the equal of laboratory assignments. If you are simply a slow typist, I've given you until the end of the

day, but ideally you should upload a file, even a non-working one, by the end of the class period and be finished with it.

- **Help One Another!** If your neighbor is struggling and you know what to do, offer your help. Don't "do the work" for them, but advise them on what to type or how to handle things.
- **Review the Honor Code policy on the syllabus.** Remember that you may discuss programs with others, but programs that are nearly identical to others will be taken as evidence of violating the Honor Code.