CS 127A (Spring 16): Introduction to Computer Science

Section Activity #4
Scanner

What’s the Point?
We’ve seen Scanner in class - and now, it’s time to practice using it. Not only will we be reading from the keyboard - it’s also a good chance to get more practice with loops.

What We’ll Be Doing
Here are the key methods and techniques we’re going to practice:

- Creating a Scanner object
- Using hasNext() and next()
- Modifying variables in a loop

Pair up!
Remember, we’ll be doing pair programming this semester. So choose a partner (somebody you’ve never worked with in section), and find a computer.

Part 1: Pseudocode
We’re still experimenting with the group-pseudocode projects. For our first checkpoint, the entire Section will work together to write pseudocode for the first program.

Program 1
This program is a chatterbox. It will repeat any word that you type, any number of times that you want. It will read data from the keyboard using Scanner; you will type an integer, and then a word, and it will repeat the word that many times. The program ends when you hit Ctrl-D.

This program will not read anything from the command line; it will simply read commands from the keyboard. As a section, review slide deck B, slides 3-8, to see the key parts of a loop which reads from a Scanner.

After you’ve reviewed those slides, work on the pseudocode. The pseudocode should:

- Have a loop to read from the keyboard. End the loop when the user hits Ctrl-D.
- Read two words from the user. Use `Integer.parseInt()` to convert one of them to an `int`.

- Have a small loop, nested inside the main loop, which prints out the word the number of times requested.

- When the loop ends (because the user typed Ctrl-D), print out three values:
  - The number of words typed by the user
  - The total number of words typed by the program
  - The total number of characters (not counting newlines) typed by the program.

  How will you remember these values for when you need them at the end?

This section ends when your Section Leader says so; you may finish all of the points above, or your Section Leader may end it early.

**NOTE:** This time, I've chosen not to give you a point just for doing the group pseudocode. Instead, you’ll get it when you finish converting this program to Java.

**Part 2: Convert it to Java**

Log onto your lab computer. Open up a text editor, and copy down all of the pseudocode that the Section has written (unless your SL comes up with a faster way, that lets you download it).

Next, open up the class homepage: [http://www.cs.arizona.edu/classes/cs127a/spring16/](http://www.cs.arizona.edu/classes/cs127a/spring16/)

Once you’re there, open up two things:

- This section activity
- Slide deck 04 (“Loops”)
- Slide deck B (“Scanner”)

Now we’re ready to get started.

**Scanner**

Open up DrJava, and create a new class. Create the shell of the program (remember, you can look at Slide Deck 02, slides 42-44). Make sure that you add the `import` statement for the `Scanner` class (slide deck B, slide 4).

Now, add a line of code to create the `Scanner` object (slide deck B, slide 5). Build the `while()` loop to read from the
Java, Step 2: Convert from Pseudocode

Now, convert the pseudocode to Java, a few lines at a time. It’s wise to run your program every few lines (or at least compile it), so that you can find errors early.

When you have converted the pseudocode to Java, test your code to make sure it works. When it does, you’re done with this checkpoint!

嘀嘀嘀	CHECKPOINT 2

Raise your hand. Your SL will come over and verify that you completed this step correctly.

Time to switch!
Log out, and switch seats.

Program 2: Number Guesser

In this program, we will read an integer from the command line - and then will play a guessing game, as the person at the keyboard tries to figure out what the number is.

I’ll explain it best with a picture:
So the idea here is that the command line argument is the “target” - the number that the player is trying to guess. So you’ll need to convert the first command line argument to an integer.

Then, in the program itself, you will read numbers from the keyboard. How do you read numbers? Simple - you read words using `next()`, and then you convert them to integers with `Integer.parseInt()`!

Each time that you read a number, you will compare it to the target - if it is equal, then say so; if it is too low or too high, then tell the user that. (If the user doesn’t get it right, then also remind them exactly what the target and the
(Note that my program has a bug - it keeps asking you for guesses, even after you’ve figure out the number. So hit Ctrl-D to end the program. Later in the semester, I’ll show you how to terminate a loop with the \texttt{break} statement.)

By the way, I want you to print out the line

\texttt{target=10\ guess=123}

exactly like you see it in the example picture. This will require that you chain a bunch of things together in a single \texttt{println()} statement. Remember, I’ve told you in lecture that you can chain many things simply by adding more plusses!

Additionally, I want you to show your Section Leader that you know how to use Ctrl-D to end keyboard input.

\textbf{CHECKPOINT 2}

Raise your hand. Your SL will come over and verify that you completed this step correctly.

\textbf{Time to switch!}

Time to switch seats! The driver becomes the navigator, and vice-versa.

\section*{Program 3: Remembering the Previous Word}

In this program, you will read many words from the command line; each time you read a new word, you will print out both the \texttt{previous} word, plus the new one.
The trick with this program is that you need to declare a variable which you will update (and read) in each pass of the `while()` loop. To help you out, I’ll give you the first few lines of the `main()` method:

```java
Scanner in = new Scanner(System.in);
System.out.println("TYPE MANY WORDS, ONE PER LINE:");
String prev = in.next();
while (in.hasNext())
```

These first few lines create the `Scanner` object, print out a prompt, and then read a single word from the keyboard. We then enter a `while()` loop, which will keep running until the user hits `Ctrl-D`.

Your job is to finish the code for `main()`. What do you need to do in each pass of the loop? How do you remember the previous word that you read from the keyboard?

✓ CHECKPOINT 3
Raise your hand. Your SL will come over and verify that you completed this step correctly.

The End: Clean Up

The G/S 930 lab is host to sections from several of our programming classes, and is available the rest of the time for any CS student to use to work on their programming assignments. To help you get in the habit of leaving your work space neat and clean, we’ll end each section activity with this checkpoint, which also serves as confirmation that you attended section - even if you get nothing else done, you can clean up before you go, thereby earning one checkpoint and proving you attended.

- Log out of your computer.
- Pick up your papers, writing implements, cell phones, etc.
- Push in your chairs.

✓ CHECKPOINT 4

Raise your hand. Your SL will come over and verify that you completed this step correctly.

As with so much in this course, thanks to Dr. McCann for some of the ideas - and even some of the exact text - for this Section Activity!