COLBY COLLEGE
Department of Mathematics
and Computer Science

CS115 **Allen B. Downey**
Spring 1999

Assignment 7: Strings and things

Due: Wednesday 31 March

Experiments!

In lecture I talked about various ways that `char`s, `String`s and `int`s get combined, and we made a couple of guesses about what Java does.

1. Create a new project named Test (no graphics).
2. In `main`, write some simple expressions that combine various types using the `+` operator. For example, what happens when you "add" a `String` and a `char`? Does it perform addition or concatenation? What is the type of the result?
3. Make a big copy of the following table and fill it in. At each intersection, you should indicate what operation is performed (addition or concatenation), and what the type of the result is.

	boolean	char	int	String
boolean				
char				
int				
String				

4. Think about some of the choices the designers of Java made when they filled in this table. How many of the entries seem reasonable, as if there were no other reasonable choice? How many seem like arbitrary choices from several equally reasonable possibilities? How many seem stupid?

   You don't have to write answers to any of those questions, but you can if you want.

Palindromes

1. Create a new (no graphic) project named Palindrome.
2. Write a method named `first` that takes a String and returns the first letter. Test this method.
3. Write a method named `last` that takes a String and returns the last letter. Test this method.

4. Write a method named `middle` that takes a String and returns a substring that contains everything *except* the first and last characters.

   Hint: use the substring method. Here is the documentation:

   ```
   public String substring (int beginIndex, int endIndex)

   Returns a new string that is a substring of this string. The substring
   begins at the specified beginIndex and extends to the character at
   index endIndex - 1.

   Parameters:
   beginIndex - the beginning index, inclusive.
   endIndex - the ending index, exclusive.

   Returns: the specified substring.
   ```

   Try to figure out what they mean by "inclusive" and "exclusive." Run a couple of experiments to see if you are right.

5. A palindrome is a word that reads the same both forward and backward, like "otto" and "palindromeemordnilap." An alternative, recursive definition of a palindrome is "either an empty string or a word whose first letter is the same as the last, and whose middle is a palindrome."

   Write a recursive method named `isPalindrome` that takes a `String` and that returns a boolean indicating whether the word is a palindrome or not.

6. On a piece of paper, figure out a strategy for checking palindromes iteratively. Think about what variables to use and what role they will play. Show me your plan before you proceed.

7. Once I have ok'ed your plan, write it up in Java.

Names

When names are recorded in a computer, they are sometimes written with the first name first, like "Allen Downey," and sometimes with the last name first and a comma, like "Downey, Allen."

In order to compare names and put them in alphabetical order, it is necessary to convert them into a common form.

1. Create a new project (no graphics) named Name.

2. Write a method called `hasComma` that takes a name as an argument and that returns a boolean indicating whether it contains a comma. If it does, you can assume that it is in last name first format.

3. Write a method called `convertName` that takes a name as an argument. It should check whether it contains a comma. If it does, it should just return the string.

   If not, then it should assume that the name is in first name first format, and it should return a new string that contains the name converted to last name first format.

Documentation

1. Go to the following web page:

   `http://java.sun.com/products/jdk/1.1/docs/api/packages.html`

2. Click on `java.lang`. This is the package that contains many of the built-in Java classes.

3. Click on `String`. A pretty long page will appear containing some general information about Strings along with documentation of *all* the String methods. Print this page and add it to your three-ring binder of class materials. Highlight the methods we have used so far.

What to turn in

1. Please print and prepare your palindrome code as usual. Also, turn in a copy of the addition table. You do not have to answer the questions above in writing, but you should think about them.