Faculty of Arts & Sciences
Department of Computer Science
CMPS 200—Introduction to Programming
Assignment 9 – Due Monday Nov 26, 2012

Notes and Announcements

- Reading material: pp 497-514 from chapter 7 of the text (2D arrays)

Exercises

1. PasswordChecker

Write a program `PasswordChecker.java` that reads a string interactively from the user and checks whether it is a "good" password. Assume "good" means that it (i) is at least 8 characters long, (ii) contains at least one digit 0-9, (iii) contains at least one upper case letter, (iv) contains at least one lower case letter, and (v) contains at least one non-alphanumeric character. Your program should keep asking the user to enter a password until a “good” one is entered.

2. WordLengths.

Write a program that takes the name of input file as command line argument. The program should count the number of letters in each word of the file (a “word” is any sequence of printable characters; this is also formally called a “token”) and print a histogram of how many word lengths (how many contain each number of letters). Your program should print the histogram in two ways: on the console (as shown on page 522 of your text) or in a drawing window using vertical colored bars.

3. Day of the Year

Write a program that reads in the name of a month (3 letter abbreviation) and the day of the month as command line arguments, and prints the number of that day (a number between 1 and 365). Assume that the year is not a leap year. Use the following template.

```
public static int dayOfYear (String month, int day) {
    String[] months = {
                        "Jan", "Feb", "Mar", "Apr", "May", "Jun",
                        "Jul", "Aug", "Sep", "Oct", "Nov", "Dec"
                      };
    int[] days = {
                    31, 28, 31, 30, 31, 30,
                    31, 31, 30, 31, 30, 31
                 };
    // insert code here
}
```

4. Array Operations

Write the following methods that operate on 2D arrays.

// generate and return an array of doubles with n rows and m columns filled with random numbers from 0 to 1
```
public static double[][] generate2D(int n, int m)
```

// generate and return an n by m array of doubles filled with random numbers between min and max
```
public static double[][] generate2D(int n, int m, int min, int max)
```

// returns the sum of all the entries of a 2D array
```
public static double sum(double[][] a)
```

// returns the sum of the entries that are in the lower part of the 2D array
```
public static double sumLowerBlock(double[][] a)
```

The lower block of an array consists of the entries whose row numbers are greater than or equal to their column numbers.

Write a `main()` method that calls the methods above to test them.

5. Transpose

Write a program `Transpose.java` that reads a 2D array from a file and prints its *transpose* (rows and columns interchanged) to standard output.

Your program should consist of the following methods:

```
// Create and return a 2D array read from standard input
public static int[][] readArray(String filename)

// Print a  2D array to  standard output
public static void printArray(int[][] a)

// Transposes a 2D array in place
public static void transpose(int[][] a)
```

The input file that your program should read consist of two numbers representing the number of rows (n) and number of columns (m) in the array, followed by $n \times m$ integers that represent the entries. For example, consider the file matrix1.txt containing the following:

```
4 3
12 0 18 -1 2 3 7 8 0 -12 8 1
```

An invocation of the program using this input file might produce:

```
> java Transpose matrix1.txt
12 -1 7 -12
0  2  8  8
18 3  0  1
```

6. Answers Consider a True/False quiz that contains five questions. Suppose that the file `answers.txt` contains student answers with each line representing the responses of a particular student. The input file `answers.txt` consist of one number x representing the number of the students, followed by x rows that represent the student answers. For example, consider the file answers.txt containing the following:

```
3
T T T T F
F F T T T
F T T T T
```

`Suppose also that` the input file `correct.txt` contains the correct answers. For example, consider the file correct.txt containing the following:

```
F F T T T
```

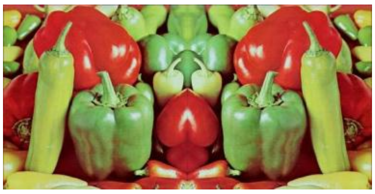
Assume that each question is worth 4 points. Write a program `Answers.java` that calculates the students' scores on the quiz and store these computed scores in a one-dimensional array.
An invocation of the program using these input files might produce:

```
> java Answers answers.txt correct.txt
Student 1: 8
Student 2: 20
Student 3: 16
```

Hint: read a 2D and 1D array of characters from answers.txt and correct.txt, respectively.

7. Mirror

Write a program to read in a W-by-H picture and produce a 2W-by-H picture that concatenates the original W-by-H picture with its mirror image of the W-by-H picture. Use the peppers.jpg image (available on Moodle) to test your program.



8. TicTacToe with 2D arrays

Redo the TicTacToe program from last week using a 2D array representation of the board.