

## Homework #1 Introduction to Computing

### Due: Sept. 5, 2009 (Saturday at 11:59 PM)

For homework #1 you are to write two programs. The first program is the interactive **Word Game** program described in Chapter 3 Programming Challenge 25. (page 161). (**I've repeated it below.**) The second program is exactly the same, except it gets its input from a text file (storyData.txt) and writes its story to an output file (story.txt).

**First program details:** Name the project folder WordGameInteractive

25. Write a program that plays a word game with the user. The program should ask the user to enter the following:

- His or her name
- His or her age
- The name of a city
- The name of a college
- A profession
- A type of animal
- A pet's name

After the user has entered these items, the program should display the following story, inserting the user's input into the appropriate locations:

```
There once was a person named NAME who lived in CITY. At the age of
AGE, NAME went to college at COLLEGE. NAME graduated and went to work
as a PROFESSION. Then, NAME adopted a(n) ANIMAL named PETNAME. They
both lived happily ever after!
```

**Second program details:** Name the project folder WordGameFileProcessing

Copy your above program and modify it so that it gets its input from a text file (storyData.txt) and writes its story to an output file (story.txt). You can create the storyData.txt file by going to the Solution Explorer tab in Visual Studio and right-clicking on Source Files. Then, select Add | New Item and from the "templates" pane select Text File (.txt). Enter the name storyData. The storyData.txt file will be created in the project folder WordGameFileProcessing under the subfolder WordGameFileProcessing. This allows your program to open the file without a path name, i.e., `infile.open("storyData.txt");`. When the program runs, it will create the story.txt file in this same subfolder.

When you write your programs, be sure to use general conventions of good style:

- use meaningful variable names with good style, i.e., use CamelCase (or use underscores) (I like to declare them one per line with a following comment if necessary)
- use meaningful named constants (e.g, PI, STATE\_SALES\_TAX) where appropriate with good style (ALL\_CAPS\_AND\_UNDERSCORES). Put your global constants where they can be found and changed easily in future versions of your program, e.g, after your initial comments describing the program and before your main function definition.
- use comments at the start of the program, before each function, and before any especially difficult section of code to understand (I like to label the closing set bracket, '}', with some indication of what's being closed)
- place the main function near the top of the program with user-defined functions below it
- use *white space* (spaces, indentation, blank lines) to make you program more readable by:

- aligning the opening set/curly brace, '{', with the corresponding closing '}' one (I like to put the '{' on the same line with the programming construct (e.g., main function definition) with the closing '}' aligned with the start of the construct)
- indent all the lines inside a set of of braces
- you blank lines to separate logical units of the code, e.g., between variable declarations and executable statements

**Submit your homework electronically at [http://math-cs.cns.uni.edu/~schafer/submit/which\\_course.cgi](http://math-cs.cns.uni.edu/~schafer/submit/which_course.cgi)**

The steps for the homework submission system are:

1. Zip your project folder WordGameInteractive by right-clicking on it and selecting Send To | Compressed Zipped Folder which will create a file called WordGameInteractive.zip. Similarly, zip your project folder WordGameFileProcessing to a file WordGameFileProcessing.zip. You want to submit both of these via the on-line submission system.
2. Log on to the submission system at: [http://www.cs.uni.edu/~schafer/submit/which\\_course.cgi](http://www.cs.uni.edu/~schafer/submit/which_course.cgi)  
(It is very likely that you will get some security certificate warnings when trying to use this. You may add an exception and accept the existing security certificate.) Use the same AD-ITS User name and password you use to log on the lab computers.
3. Select the course and section number of "810:051, Intro to Computer, Fienup". Click the "Continue" button.
4. Select the homework that you wish to submit: "HW 1 Word Game". Click the "Continue" button.
5. Specify how many extra files you want to submit. Just leave it at 0. Click the "Continue" button.
6. Upload each .zip file by Browsing and selecting .WordGameInteractive.zip and then WordGameFileProcessing.zip. Click the "Continue" button.
7. The next page reports on the status of the upload(s). You can always continue to upload until the deadline with a newer file of the same name replacing the older one.