Assignment One

SoftEng206, Semester 2, 2013

Due: Monday 12^{th} August, 11:59pm

Introduction

This assignment requires that you translate two given C programs into implementations of BASH and Java. Objectives:

- Develop your journal keeping skills.
- Expose you to the process of (rapidly) applying Software Engineering fundamentals to unfamiliar languages.
- Become comfortable using a UNIX-like environment and tools.

Assessment

- This assignment is worth 10% of your course mark. All files will be submitted via the Cecil dropbox.
- All programs will be executed on the ECE Remote Linux image. Be sure to test *all* your work using the command line, as all programs will be executed this way. You will lose a lot of marks if the markers cannot get your program to run on Linux, this is especially important if you developed your code via Cygwin or the likes.
- You can assume that all program input will be valid, we will *not* assess corner-cases or input validation.

Task 1. DataTask: Working file data

Managing data may be difficult, especially when they are stored in different files. Scripting in BASH may be more powerful when dealing with certain types of problem compared to a general purpose programming language such as Java.

The provided `DataTask.c` identifies all files in the current directory whose file name contains "contact". It will read the content of these files, sort them in alphabetical order and write the output into a file called `out.txt` in the current directory. For simplicity, the content of these files will only contain a single lower-case word.

1. Compile and run the program `DataTask.c` and document your findings in your journal. This could include instructions or options used to compile the program, how you executed it, what output was produced, whether you could find any flaws, and the identification of any key features in the program or it's implementation (the source code).
2. Develop a Java implementation named `DataTask.java`, then compile and run it at the shell prompt: `> java DataTask`.
3. Develop a BASH implementation named `DataTask.sh`. Ensure this file is executable so that a user can run your program like this: `> ./DataTask.sh`.
4. Document in your journal your experiences when working with the two languages. Which one was simpler to implement? What are some of the key things that you've learnt / found difficult? Would you use BASH again for a similar situation? Why or why not?

Task 2. FileTask: Working with files

Files are an important part of many operating systems such as Windows or Linux. It is important to become familiar with how to manipulate files using code; a good automated process may avoid the tedious process of doing it manually.

The provided `FileTask.c` will take in three inputs (Directory, Contact Name and Contact Number) through either standard input or command-line arguments. You may assume:

1. Directory has no spaces (but make sure cases like ".", "myDirectory" and "/myDirectory" works).
2. Contact Name is only a single word consisting only of lower case letters.
3. Contact Number only contains digits.

The program will create one file in the specified Directory and name it in the format `<Contact Name>.contact`, for example `john.contact`. The content of the file is the Contact Number. If for example `john.contact` already exists, `john(2).contact` will be used, then `john(3).contact` and so on. Repeat all steps in Task 1, but now for this new program (`FileTask.c`) and name your new implementations accordingly. Hint: make sure you understand the difference between standard input and command line arguments and document this in your journal. Also comment on how users may use pipes "|" or redirection "<" to feed input into your program.

Task 3. Language evaluation

Write a short journal entry discussing and contrasting the differences in your implementations for Tasks 1 and 2. Think about the strengths (if any) and weaknesses (if any) of the languages you have used. How did you find the learning process, can you give general advice to a person learning either of these languages? You are basically submitting your journal entries, but as an electronic version (either scan the respective pages from your journal, or use some other journal software).

Resources

We will introduce each of the languages in lectures, and give you enough information to get started. But you will likely have to do some of your own research and find some resources yourself. The following web sites are good to get started:
`http://www.cplusplus.com/reference/clibrary/`
`http://en.wikipedia.org/wiki/List_of_Unix_utilities`
`http://tldp.org/LDP/Bash-Beginners-Guide/html/`
`http://google.co.nz`
Any relevant textbooks from the library