COMP 30400 – Programming I & II
Lab 7 – 2 hours

Practical Overview

This practical is for Lecture 7. All portfolio programs require a header at the start of every source file. Refer to lecture notes on how to format it. Comments are expected throughout the code, where relevant. They should be used to discuss motivations for your approach and alternative ways of implementing the solution where appropriate.

Appropriate error checking for memory allocations, and free-ing of memory and files is required where relevant.

Be careful not to accidently delete files when using fopen(…, "w");

1) Portfolio Program 31
Based on Program 19 (3 chances to guess a word), write a program that reads the secret word from a text file. The rest of the program should remain the same (i.e. the user still gets 3 chances to guess the word.

2) Portfolio Program 32
Write a program that can copy a text file:

FileCopy inputfile outputfile

Hints: Remember to use getc() and putc(). Also recall that you will need to use the correct file modes – "r" for reading the file, and "w" for writing the file.

3) Portfolio Program 33
Based on Program 31, modify the program so there are 20 secret words in the text file. If the user guesses the word correctly, the program should start using the next secret word in the text file, and give the user 3 guesses again.

4) Portfolio Program 34
Write a program that compares two files, and tells the user if they're identical or not. The user can specify the two filenames on the command line. Example:

FileCompare inputfile outputfile

5) Portfolio Program 35
Write a program to display file contents 20 lines at a time. The program pauses after displaying 20 lines until the user presses either Q to quit or Return to display the next 20 lines. (The Unix operating system has a command called **more** to do this) As in previous programs, we read the filename from the user and open it appropriately

6) Portfolio Program 36
Write a program that inputs a .c source file, and counts the number of brackets in the file. The program should alert the user if the amount of {,(,['s does not match the

amount of },),] ‘s respectively. The user should be able to specify the filename of the .c file on the command line. For example:

```
checkbrackets.exe mysource.c
```

7) Remember to backup your portfolio.