G51PRG Exercise Three:
Files and Arrays: Finding the Largest Pair

Steven R. Bagley

Introduction

For this coursework, you are to implement a computer program which finds the pair of integer values with the highest total value, from a list of ten integers. However, the total must also be equal to or below a threshold of 50[†].

For example, consider this list:

```
10, 25, 13, 44, 9, 15, 6, 27, 36, 42
```

The highest valued pair with a total less than the threshold is 44 and 6, which gives a total of 50. There are other pairs with a higher total (e.g. 25 and 42), but we are only interested in pairs with a total of less than or equal to 50.

You are to create two versions of this program. The first will read in a set of ten integers from a file into an array and find the largest pair of numbers. The file may contain more than one line of numbers and you should process each line separately until you reach the end of file. The second version of the program will read the entire file into a *two dimensional* array and find the line (or lines) containing the largest pair. It's probably best to write the first version first, and then adapting it to handle the second problem.

1. Finding the largest pair per line

Your program should read each list of integers from a file, and store the values in an array. You can assume that the file contains only complete lists (one per line), and that each list contains exactly ten integers. For each list of ten integers, you must loop through every possible pair of numbers, e.g. using two nested `for` loops.

As you process each pair, you must keep track of the maximum total that you have found so far, checking that the total is less than or equal to the threshold. The threshold for all tests is 50 (so pairs that total 51 or greater do not count, and should not be stored as the highest). When you search through each pair, be careful that you do not compare the same number to itself. In the example above, 25 and 25 makes 50, but there are not two separate values of 25 in the list. The pair must be made up of two distinct values.

The file is in this format, where each row represents one list of ten integers:

```
 4, 15, 47, 23, 18, 10, 22,  6, 37, 28
40, 19, 17, 23,  2, 43, 35, 21,  4, 34
 4, 25, 41, 48, 34, 20, 10, 19,  7, 16
39,  6, 10, 34,  8, 17, 23, 49, 38, 12
```

To read in the values you should use the `fscanf()` function. Your program should accept the name of the file to be opened (using `fopen()`) as the first parameter on the command line. Therefore, if your program is called `processpairs` and the file containing the lists of integers is called `items.txt`, you would call your problem as follows (this assumes that items.txt is in the same directory as your program):

[†] This is a simple variation of the classic knapsack problem, which is very well known in Computer Science (`http://en.wikipedia.org/wiki/Knapsack_problem`).

```
./processpairs items.txt
```

Your program should output the values in this format:

```
Line 1: largest pair is 22 and 28, with a total of 50
Line 2: largest pair is 43 and 4, with a total of 47
…
```

You will need to download the text file from the web so you can use it to test your program. You can do this via Windows and save it to your `H:` drive in the correct place, or you can do it from the Linux command prompt by using the `curl`

```
curl -O http://g51prg.cs.nott.ac.uk/Distribution/Coursework/items.zip
unzip items.zip
```

There is no single correct solution for this exercise; any program which correctly calculates the answers will be given a pass mark. To achieve higher grades, you should try and make sure your program doesn't do any calculations it doesn't need to. Also, to make it easier to implement the second half of the coursework you may want to think about how you use functions and which parts of the program would make sense to be in a separate function.

2. Finding the largest pair in the file

This version of the program is very similar to the first, the difference is that it only the lines containing the largest pair in the file are printed out. So for the sample file `items2.txt`, the output for this version of the program would be the following lines:

```
Line 4: largest pair is 45 and 3, with a total of 48
…
```

To do this, you will need to read the whole file into a two-dimensional array (you can assume that there will not be more than six lines in the file) and then process each row of that array to find the largest pair total on that row. You'll need to keep track of the maximum pair total across all rows in the array, and then print out each row that has a pair that matches that pair total.

Again, to achieve high marks you should avoid performing unnecessary calculations. You might find it helps to use other arrays to keep track of the pair totals as you process the two-dimensional array.