

# CS4222 Software Development Assessment

## Part 4 Specification

### Introduction

The goal of this exercise is to explore how Java “Map” structures might be used to develop a Java program that processes a file of data about course applicants. The task could be described as a “maintenance” exercise in that we are given an existing piece of code and asked to modify it, or in this case, add components to complete it. So part of the exercise involves understanding how the existing code works, and part of it requires us to write the additional code and incorporate it into the solution.

This exercise is similar to the *AnalyseTweets.java* sample program circulated in Week 11. You may find it useful to review that sample program before completing this task. In addition, the other TreeMap sample programs (e.g. Scrabble.java) might provide helpful guidance on how to complete this exercise.

For this exercise we are once again making use of a Comma Separated Values (CSV) file ([https://en.wikipedia.org/wiki/Comma-separated\\_values](https://en.wikipedia.org/wiki/Comma-separated_values)). Each line of the file is an entry about an applicant. Each line consists of one or more data items separated by commas.

The CSV file stores data about applicants to a fictitious university course. A sample file looks like this

```
21458695,H3,H6,O8,H3,H8,H1,H4,H1,  
21929116,O4,O8,O7,H7,O8,H3,H3,H5,H6  
21337254,H8,H3,H8,H7,H7,H4,,,  
21127094,O4,O3,O2,H3,O6,O5,H4,,  
21821974,H3,O5,O2,H3,H8,H4,H2,H7,H7
```

Each line in the file has data about one applicant. The data in each line is formatted so that the first value is the applicant’s identification number and this is followed by a MINIMUM of 6 and a MAXIMUM of 9 exam subject grades, each separated by a comma.

We want to calculate the applicants “points score” and decide if they are candidates for the course. To calculate the score each grade is awarded a number of “points” using the following table

Higher Level Grade	Points	Ordinary Level Grade	Points
<b>H1</b>	<b>100</b>		
<b>H2</b>	<b>88</b>		
<b>H3</b>	<b>77</b>		
<b>H4</b>	<b>66</b>		
<b>H5</b>	<b>56</b>	<b>O1</b>	<b>56</b>
<b>H6</b>	<b>46</b>	<b>O2</b>	<b>46</b>
<b>H7</b>	<b>37</b>	<b>O3</b>	<b>37</b>
<b>H8</b>	<b>0</b>	<b>O4</b>	<b>28</b>
		<b>O5</b>	<b>20</b>
		<b>O6</b>	<b>12</b>
		<b>O7</b>	<b>0</b>
		<b>O8</b>	<b>0</b>

If an applicant has more than 6 grades then only the best 6 (i.e. the six with the highest points value) are used to calculate the points total.

A partially complete version of the program has been provided in the Assignments folder on SULIS. The file is called "ApplicantAnalysis.java" and it includes the code to read the CSV file. However, the following methods have to be added

```
static  
LinkedList<String>  
select(TreeMap<String,Integer> candidateScores,  
int cutoff)
```

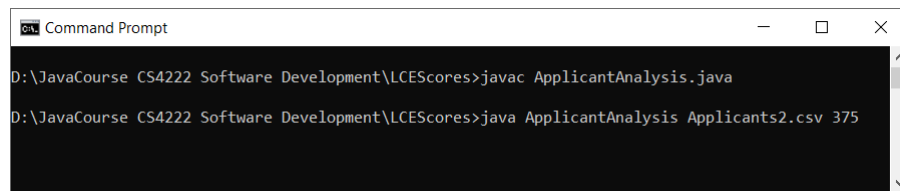
The method steps through the entries in the TreeMap and creates a LinkedList of applicant numbers by adding the applicant numbers of applicants whose score is greater than or equal to ( $\geq$ ) the cutoff score.

```
static int  
pointsScore(String[] subjectGrades) {
```

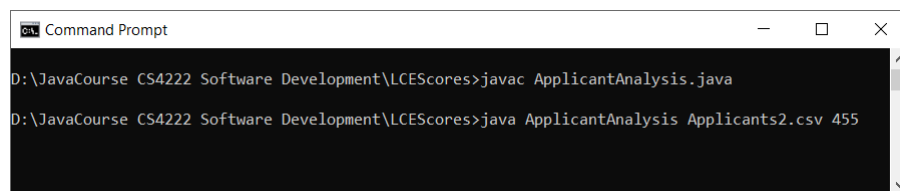
Determines the points associated with each of the grades in the subjectGrades array and calculates the total points score for the grades. If the subjectGrades array has more than six grades then only the best six (i.e. the six with the highest points) should be included in the total. The total is returned as the result of the calculations.

To use the program you can use the command line or the BlueJ method dialog box.

Once the program has been compiled you can use the command line as follows



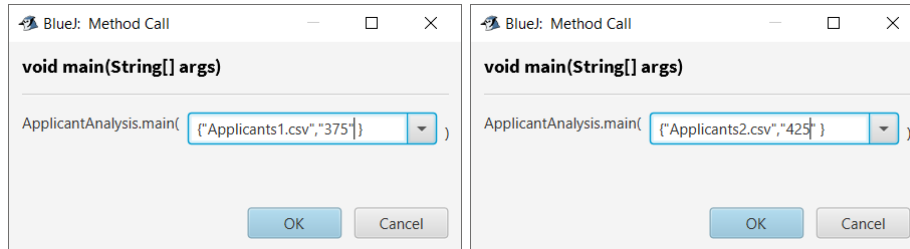
```
Command Prompt  
D:\JavaCourse CS4222 Software Development\LCEScores>javac ApplicantAnalysis.java  
D:\JavaCourse CS4222 Software Development\LCEScores>java ApplicantAnalysis Applicants2.csv 375
```



```
Command Prompt  
D:\JavaCourse CS4222 Software Development\LCEScores>javac ApplicantAnalysis.java  
D:\JavaCourse CS4222 Software Development\LCEScores>java ApplicantAnalysis Applicants2.csv 455
```

So, after the “java ApplicantAnalysis” part of the command you have to specify the name of the CSV file AND then the integer value for the cutoff. If you are using a filename that includes spaces then you should put the filename inside quotation marks.

If you prefer to use the BlueJ dialog box then type the parameters inside the {} and enclose BOTH of them inside quotation marks with a separating comma between them as follows



## Submission Requirements

Your solution to the assignment should be submitted on Sulis on or before **16h00 on 27th April 2022**. Submissions will be accepted up to **23h55 on 27th April 2022**.

You should submit **ONE** file called **ApplicantAnalysis.java** that implements the operations specified for the program. Include prominent comments that contain your ID number and your Name. Sulis will accept any filenames you use but it is extremely important that you adhere to the file name conventions.

You should NOT submit zip files.

**This part of the assignment is worth 25% of the total of 100%.**

**The marks will be allocated as follows**

Method	Allocation
pointsScore	15
select	10
Total	25