

# Seneca College

Applied Arts & Technology  
SCHOOL OF COMPUTER STUDIES

**JAC444****Submission date:****June 23, 2020**

## Workshop 4

### Description:

The following workshop lets you practice basic java coding techniques, creating classes, methods, using arrays, **Java I/O**, inheritance, polymorphism, Exceptional Handling.

### Task 1:

Write a program for *hangman game* that randomly picks a word from a text file named **hangman.txt** and prompts the user to guess one letter at a time, as shown in the sample run.

- Each letter in the word is displayed as an asterisk. When the user makes a correct guess, the actual letter is then displayed.
- When the user finishes a word, the following steps should happen
  - Display the number of misses to the user.
  - Ask the user to give you a new word, then add that word to your file **hangman.txt**. (Optional if you want to check for the duplicate words in the file)
  - Ask the user whether to continue to play.
- If the user missed the same letter more than one time give extra hint to the player like “You’ve already tried this letter, try another letter”. Don’t count twice the same letter which is missed by the user.
- Have minimum 10 words of your choice in the file **hangman.txt**.

```
(Guess) Enter a letter in word ***** > p
(Guess) Enter a letter in word p***** > r
(Guess) Enter a letter in word pr**r** > p
      p is already in the word
(Guess) Enter a letter in word pr**r** > o
(Guess) Enter a letter in word pro*r** > g
(Guess) Enter a letter in word progr** > d
      d is not in the word
(Guess) Enter a letter in word progr** > m
(Guess) Enter a letter in word progr*m > d
You have already tried d, try a new letter
```

```
(Guess) Enter a letter in word progr*m > a
The word is program. You missed 1 time
Enter a new word to be added in the memory> apple
Do you want to guess another word? Enter y or n>
```

**Note:** Students can make the output better as well in easier and more readable format.

## Task – 2

Write a program that prompts the user to enter a file name and displays the occurrences of each letter in the file. Letters are case-sensitive. Here is a sample run:

```
Enter a filename: Somefile.txt
Number of A's: 56
Number of a's: 24
Number of B's: 134
...
Number of Z's: 9
Number of z's: 2
```

## Workshop Header

```
/*****
```

**Workshop #**

**Course:**<subject type> - Semester

**Last Name:**<student last name>

**First Name:**<student first name>

**ID:**<student ID>

**Section:**<section name>

*This assignment represents my own work in accordance with Seneca Academic Policy.*

*Signature*

**Date:**<submission date>

```
*****/
```

## Code Submission Criteria:

Please note that you should have:

- Appropriate indentation.

- Proper file structure
- Follow java naming convention
- Document all the classes properly
- Do Not have any debug/ useless code and/ or files in the assignment
- Do not have everything in the *main method*.
- Have a separate TestClass with the main method in it.
- Check your inputs if the user is not entering garbage inputs.
- Use exceptional handling or other methods to let the user know if the inputs are incorrect.

## Deliverables and Important Notes:

**All these deliverables are supposed to be uploaded on the blackboard once done.**

- You are supposed to create video/ record voice/ detailed document of your running solution. **(50%)**
  - Screen Video captured file should state your last name and id, like Ali\_123456.mp4 (or whatever the extension of the file is)
  - Record voice clip should also include a separate word file with the screen shots of your program's output, state your last name and id, like Ali\_123456.mp3 (or whatever the extension of the file is)
  - Detailed document should include screen shots of your output, have your name and id on the top of the file and save the file with your last name and id, like Ali\_123456.docx (or whatever the extension of the file is)
- A word/ text file which will reflect on learning of your concepts in this workshop. Also include the instructions on how to run your code. **(30%)**
  - Should state your Full name and Id on the top of the file and save the file with your last name and id, like Ali\_123456.txt
- Submission of working code. **(20%)**
  - Make sure you follow the **"Code Submission Criteria"** mentioned above.
  - You should zip your whole working project to a file named after your Last Name followed by the first 3 digits of your student ID. For example, **Ali123.zip**.
- Your marks will be deducted according to what is missing from the above-mentioned submission details.
- Late submissions would result in additional 10% penalties for each day or part of it.
- Remember that you are encouraged to talk to each other, to the instructor, or to anyone else about any of the assignments, but the final solution may not be copied from any source.