Task 1:

Text

Description automatically generated

Graphical user interface, text, application

Description automatically generated

The hangman program has a while loop that will keep the program running as long as the “go” variable is ‘y’ or ‘Y’ and go variable is initialized as ‘y’, which means the program will run at least once and will keep looping until “go” is set to another character. I used an array list to hold all the words in the hangman.txt file because it is resizable which will be useful when new words are added to the list. Random is used to pick a word from the list for the player to guess and the program will create a string called “hidden” that consists of several ‘\*’ equal to the length of the word. When the user enters a correct letter the “hidden” string will replace ‘\*’ with the correct letter. It will continue doing so until the “hidden” string is the same as the answer while giving player feedback like whether the letter they just entered had been entered before. Finally at the end it will ask player to enter a new word to add to the list, it will also write that new word in the hangman.txt file. It will not add it to the list if it exists already.

Task 2:

Text

Description automatically generated with medium confidence

A picture containing text, person

Description automatically generated

The program creates an array of characters that consists of both upper and lower cases of all alphabets, which is used later to compare to the text file. It also creates an array of integers to keep track of how many times the letter showed up in the text file. The “current” variable is an integer because that is what BufferedReader will return and since Java can compare a char to int, I do not need to cast the BufferedReader. Then the program will compare the current character the BufferedReader is reading to the list of characters in the “letters” array, when it finds a match, it will add 1 to the respective integer in the “count” array. Finally it will print the letters with their respective count.