Progress Report

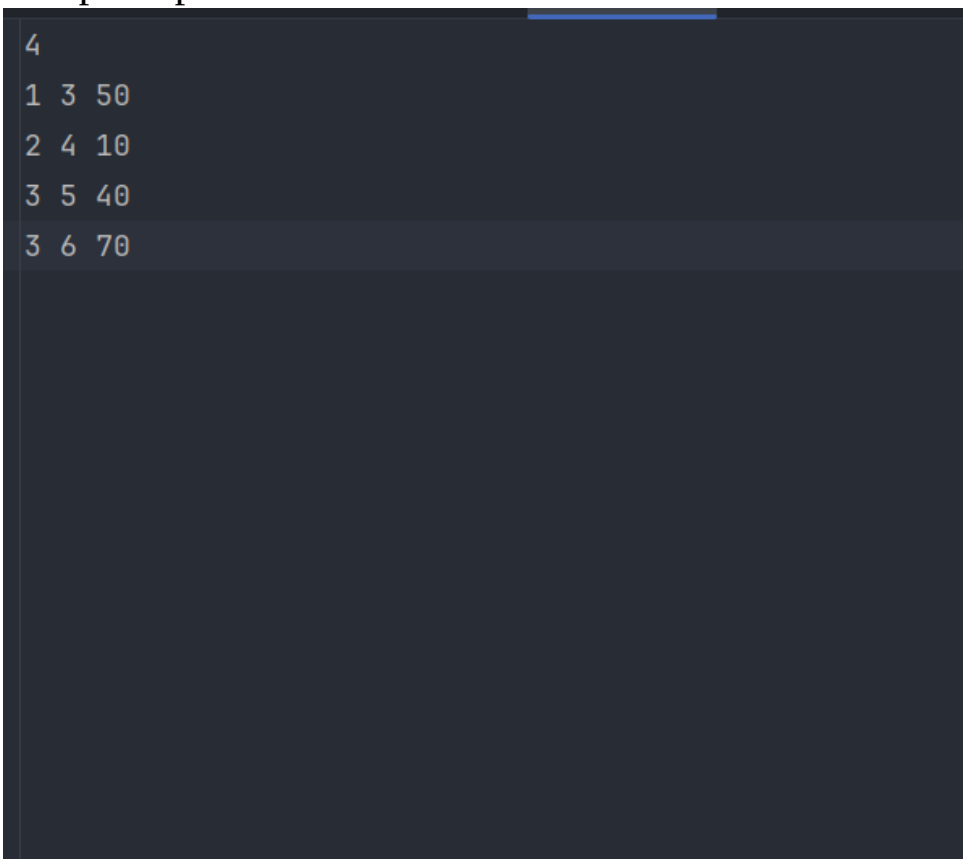
Mohamed Elsayed Amin 20011502

I acknowledge that I am aware of the academic integrity guidelines of this course, and that I worked on this assignment independently without any unauthorized help.

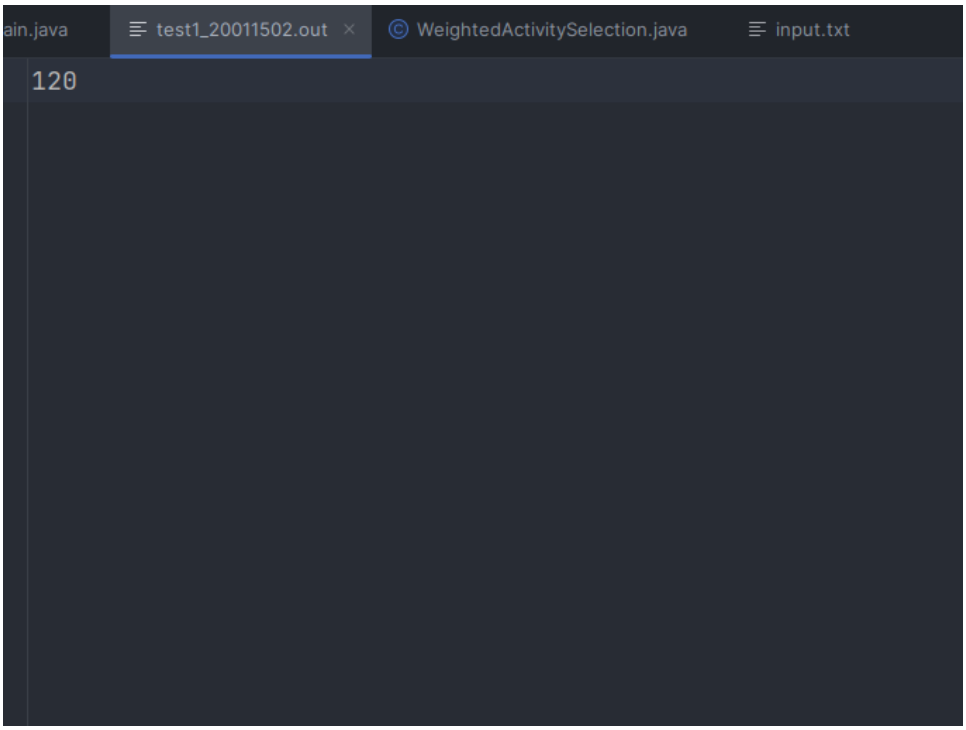
Part 1 - Weighted Activity Selection (20%)

The problem is solved in O(n log n)

Sample input :



Sample output :



Running JAR:

```
mohamed@MAmin:~/CSED_25/Year_3/Algo/My_Sheets/sheet_7_java$ java -jar activity_20011502.jar
The format should be : activity_20011502.jar absolute_path_to_input_file

mohamed@MAmin:~/CSED_25/Year_3/Algo/My_Sheets/sheet_7_java$ java -jar activity_20011502.jar  /home/mohamed/CSED_25/Year_3/Algo/My_Sheets/sheet_7_java/input.
txt
120
```

Part 2 - Huffman's Algorithm (80%)

Done:

- Done with the data structures and file parsing.
- Done with frequencies of n-byte word.
- Done with constructing Huffman tree.
- Done with writing the word-code List into the file.
- Testing most of code using
    - xxd -b input.in command to view binary data in bits:

```
mohamed@MAmin:~/CSED_25/Year_3/Algo/Huffman Compression$ xxd -b 'input.txt'
00000000: 01001000 01101001 01101100 01101000 01101001 01101111  Hilhio
00000006: 01101100 01101000 01101001 01101100 01101011 01101100  lhilkl
0000000c: 01101010 01100100 01100001 01110011 01101010 01101000  jdasjh
00000012: 01101001 01001000 01101001 01101100 01101000 01101100  iHilhl
00000018: 01101100 01101000                                      lh
mohamed@MAmin:~/CSED_25/Year_3/Algo/Huffman Compression$
```

- xxd -b input.in | grep -o '11011101' | wc –l command for fetching the frequency of specific byte

Github Gist