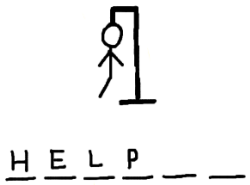
CME1212 Algorithms and Programming II
Homework 1



Upload your source code files from *DEUZEM SAKAI* until **31 March 2024, 23:55.**

Write a Java program for a simple version of ***"hangman"*** game.

"*Hangman*" is a popular word puzzle game where you guess a secret word letter by letter.

Initially

Read an unsorted file "*highscoretable.txt*" and then create two **Queues (QName and QScore)**, which are sorted by score in a decreasing order as follows:

highscoretable.txt

Kaan 90
Ali 70
Yeliz 85
Cem 100
Can 65
Pelin 75
Ece 95
Sibel 105
Ayşe 85
Berk 120
Ege 80
Ada 60

animals.txt

monkey
penguin
dolphin
umbrellabird
wildebeest
zebra
dog
...
tarantula

Example:

QName	Berk	Sibel	Cem	Ece	Kaan	Yeliz	Ayse	Ege	Pelin	Ali	Can	Ada
QScore	120	105	100	95	90	85	85	80	75	70	65	60

There is another text file "*animals.txt*" which includes 14 different animal names inside. Read the file and insert them in a stack named ***AnimalStack***. The animal names in ***AnimalStack*** is unordered.

Example:

AnimalStack	monkey	penguin	dolphin	umbrellabird	wildebeest	...	tarantula

top

All English letters should be stored in another stack (i.e. ***LetterStack***).

Example:

LetterStack	A	B	C	D	E	F	G	…	Z

top

The user has 120 points at the beginning of the game.

Start of the Game

The searched words should be the name of animals such as monkey, penguin, dolphin, umbrellabird, etc.
Generate a random number (*n*) and get n^{th} word of the *AnimalStack.*

Store this word in another stack (i.e. ***WordStack***).

Example:

WordStack	T	A	R	A	N	T	U	L	A

top

Game board should also be another stack (i.e. ***BoardStack***).

Example:

BoardStack	-	-	-	-	-	-	-	-	-

top

Game Playing

The computer specifies a word and the user tries to guess it by suggesting letters.
The word to guess is represented by a row of dashes, representing each letter of the word.

If the user suggests a letter which is found in the word, the computer writes it in all its correct positions.
If the suggested letter does not occur in the word, the computer decreases the score of the user by 15 or 20, for vowel or consonant letter respectively.
The missing letters should be stored in another stack (i.e. ***MissingLetterStack***).
You should delete the missing letter from ***LetterStack*** and insert it into ***MissingLetterStack***.

For each missing letter, you should decrease the point of the user. wovel → decrease 15 consonant → decrease 20

If the user enters the same letter a second time, an error message must be displayed and it must not decrease the points that the user has.

Joker: The user has a chance to open a letter randomly only once.

The program must display all steps until the game is over.

End of the Game

The game is over when:
- The user completes the whole word correctly or
- The user score reaches to equal or less than zero.

High Score Table

If the player earns a score within the top results, he/she will be displayed in the High-Score table.
Add into the queue Q1 and Q2. If the same score exists in the table, the new score should be inserted to the next of them. Delete the last record if it is required since the table must contain maximum 12 items.
The new score table should be written to the same file ("D:*HighScoreTable.txt*").

Play again?

At the end of the game, ask to the user if he/she wants to play again.

This homework will be graded by Res. Asst. Onur Can DOĞANLAR.
You can ask your questions her from the "**FORUM** -> **Homework 1 - Questions**" part of the *DEUZEM SAKAI* software.

Sample output:

```
Word:  - - - - - - - - -        Misses:              Score: 120    ABCDEFGHIJKLMNOPRSTUVWXYZ
Guess: A
Word:  - A - A - - - - A        Misses:              Score: 120    BCDEFGHIJKLMNOPRSTUVWXYZ
Guess: Z
Word:  - A - A - - - - A        Misses: Z            Score: 100    BCDEFGHIJKLMNOPRSTUVWXY
Guess: E
Word:  - A - A - - - - A        Misses: Z E          Score: 85     BCDFGHIJKLMNOPRSTUVWXY
Guess: Z
You entered the same letter before.
Word:  - A - A - - - - A        Misses: Z E          Score: 85     BCDFGHIJKLMNOPRSTUVWXY
Guess: S
Word:  - A - A - - - - A        Misses: Z E S        Score: 65     BCDFGHIJKLMNOPRTUVWXY
Guess: T
Word:  T A - A - T - - A        Misses: Z E S        Score: 65     BCDFGHIJKLMNOPRUVWXY
Guess: Joker
Word:  T A - A - T U - A        Misses: Z E S        Score: 65     BCDFGHIJKLMNOPRVWXY
Guess: B
Word:  T A - A - T U - A        Misses: Z E S B      Score: 45     CDFGHIJKLMNOPRVWXY
Guess: O
Word:  T A - A - T U - A        Misses: Z E S B O    Score: 30     CDFGHIJKLMNPRVWXY
Guess: I
Word:  T A - A - T U - A        Misses: Z E S B O I  Score: 15     CDFGHJKLMNPRVWXY
Guess: G
Word:  T A - A - T U - A        Misses: Z E S B O U G Score: -5     CDFHIJKLMNPRVWXY


You lost !!
Your score is -5.
```

```
What is your name: Murat

High Score Table
Berk  120
Sibel 105
Cem   100
Ece   95
Kaan  90
...

Play again?
Y


Word:  - - - - -              Misses:           Score: 120   ABCDEFGHIJKLMNOPRSTUVWXYZ
Guess: A
Word:  - - - - A              Misses:           Score: 120   BCDEFGHIJKLMNOPRSTUVWXYZ
Guess: Z
Word:  Z - - - A              Misses:           Score: 120   BCDEFGHIJKLMNOPRSTUVWXY
Guess: Joker
Word:  Z - B - A              Misses:           Score: 120   CDEFGHIJKLMNOPRSTUVWXY
Guess: U
Word:  Z - B - A              Misses: U         Score: 105   CDEFGHIJKLMNOPRSTVWXY
Guess: R
Word:  Z - B R A              Misses: U         Score: 105   CDEFGHIJKLMNOPSTVWXY
Guess: E
Word:  Z E B R A              Misses: U         Score: 105   CDFGHIJKLMNOPSTVWXY
You win !!
Your score is 105.

What is your name: Ceyda

High Score Table
Berk  120
Sibel 105
Ceyda 105
Cem   100
Ece   95
Kaan  90
...

Play again?
N
```

Notes

1- In your program, you can use the stack and queue data structures as you want, but you must use only stack and queue.
Don't use other data structures such as an array or arraylist or list.
Don't use STRING data type in the main solution, instead of a stack or queue.

2- The stack class has only the following methods: push, pop, peek, isFull, isEmpty, and size.
Don't add a new method into the stack class.

For example, don't write a *display* method in the Stack class.
For example, don't write a *search* method in the Stack class.

All other methods must be written in the ***main*** program.

3- The queue class has only the following methods: enqueue, dequeue, peek, isFull, isEmpty, and size.
Don't add a new method into the Queue class.

For example, don't write a *display* method in the Queue class.
For example, don't write a *search* method in the Queue class.

All other methods must be written in the ***main*** program.

4- You can use ***linear queue*** or ***circular queue***.

5- Don't use stack and queue classes embedded in Java. Write your own Stack and Queue classes.

6- Upload format

Step1: Create a new folder, named by your student number and name (without any space)For example: 2015510012_Ali_Tas

Step2: Copy all java files into this folder

Step3: Compress the folder **2015510012_Ali_Tas.zip**

Step4: Upload the file **2015510012_Ali_Tas.zip** from *DEUZEM SAKAI*

7- Don't use **ENIGMA** or any other extra library.

8- If you are late, your grade will be decreased by 10 points for each day. After five days, your assignment will not be accepted.

9- Assignment must be your individual work.
Cheating is strictly prohibited.
All source codes will be automatically compared with each other by using a program.If any cheating occurs, your assignment will be graded with **zero (0)**.

10- Your program must work correctly under all conditions. Try to control all possible errors.

11- You should use meaningful variable names, appropriate comments, and good prompting messages.