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.

Description about the Game

The computer randomly identifies a word and the user tries to guess it by suggesting letters, within a certain number of guesses.

The word to guess is represented by a row of dashes, representing each letter of the word.

The word should be the name of a *country* in English such as Germany, Brazil, China, etc.

If the user suggests a letter which can be found in the word, the computer writes it in all its correct positions.
If the suggested letter cannot be found in the word, the computer increases the count of guesses.

Initially

Read the <u>unsorted</u> file "D:*input.txt*" and then create a **Multi-Linked List (MLL)**, which is <u>sorted</u> by letter counts and alphabetical, as follows:
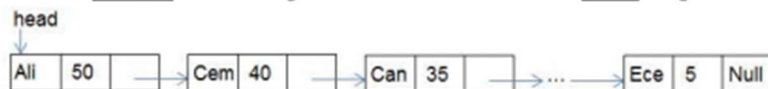
<u>Note</u>: Don't use a temporary SLL during sorting process. Insert the words from the file directly into the MLL.
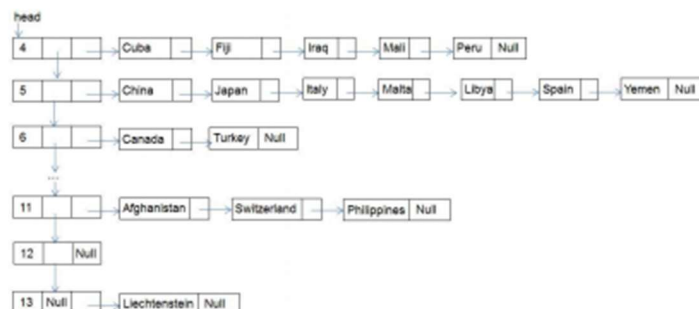
Add all letters in a **Double Linked List (DLL)**.

Read the <u>unsorted</u> file "D:*HighScoreTable.txt*" and create a <u>sorted</u> **Single Linked List (SLL)** as follows:


```
Pelin;30
Kaan;15
Ali;50
Yeliz;25
Cem;40
Can;35
Ece;5
Sibel;30
Remzi;20
Nazan;10
```

The Beginning of the Game

Generate a random number (n), print this number on the screen and get n^{th} word of the MLL.
For example; if the randomly generated number is 9, the country "*MALTA*" is selected from the aforementioned MLL.

Store this word in a **Circular Single Link List (CSLL1)**.
In addition to CSLL1, create an empty circular single list, named **CSLL2**, for the game board.

Example:
CSLL1
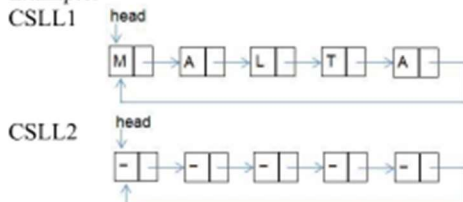
CSLL2

Game Playing
The game board (CSLL2) must be seen on the screen, permanently.

The program should continuously take a letter from the user.

In order to provide high performance, the letter given by the user should be checked in the DLL as follows:
- If the given letter is smaller than or equal to "*M*", check the DLL by starting from the head.
- If the given letter is greater than "*M*", check the DLL by starting from the tail.

The suggested letters should be deleted from the DLL.

If the suggested letter does not exist in the CSLL1, the computer increases the count of guesses.
If the user suggests a letter which exists in the CSLL1, the computer writes it in all its correct positions in the CSLL2.

For example;
If the suggested letter is "A", then the CSLL2 should be as follows:

CSLL2   Before   After
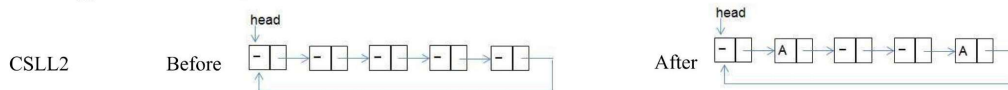
If the user enters the same letter a second time, an error message must be displayed, and it must not count against the six total guesses the user has.

The program must display all steps until the game is over. In other words, the program must continuously print the CSLL2, the number of lives, the score, and the DLL on the screen.

Scoring
The user gets 10 points for each consonant he/she knows.
The user gets 5 points for each vowel he/she knows.

End of the Game
The game is over when:
- The user completes the whole word correctly or
- The number of guesses reaches to six (6).

High Score Table
The player can record his name and the score that he/she has earned. If he/she earns a score within top-10 results, he/she will be displayed in the High-Score table which is given as text file. High Score table will contain highest 10 game scores of all time and will be stored using only a **Single Linked List (SLL)**. If the same score exists in the table, the new score should be inserted to the next of them.

Sample output:

```
Randomly generated number: 14

Word:  - - - - -          Lives: 6      Score: 0        ABCDEFGHIJKLMNOPQRSTUVWXYZ
Guess: E
Word:  - - - E -          Lives: 6      Score: 5        ABCDFGHIJKLMNOPQRSTUVWXYZ
Guess: R
Word:  - - R - E -        Lives: 6      Score: 15       ABCDFGHIJKLMNOPQSTUVWXYZ
Guess: A
Word:  - - R - E -        Lives: 5      Score: 15       BCDFGHIJKLMNOPQSTUVWXYZ
Guess: S
Word:  - - R - E -        Lives: 4      Score: 15       BCDFGHIJKLMNOPQTUVWXYZ
Guess: T
Word:  T - R - E -        Lives: 4      Score: 25       BCDFGHIJKLMNOPQUVWXYZ
Guess: B
Word:  T - R - E -        Lives: 3      Score: 25       CDFGHIJKLMNOPQUVWXYZ
Guess: Z
Word:  T - R - E -        Lives: 2      Score: 25       CDFGHIJKLMNOPQUVWXY
Guess: S
You entered the same letter before.
Word:  T - R - E -        Lives: 2      Score: 25       CDFGHIJKLMNOPQUVWXY
```

```
Guess: O
Word:  T - R - E -          Lives: 1      Score: 25              CDFGHIJKLMNPQUVWXY
Guess: L

You lost !!

High Score Table
Ali 50
Cem 40
Can 35
...
Ece 5
```

Sample output:

```
Randomly generated number: 9

Word:  - - - - -                  Lives: 6      Score: 0          ABCDEFGHIJKLMNOPQRSTUVWXYZ
Guess: A
Word:  - A - - A                  Lives: 6      Score: 5          BCDEFGHIJKLMNOPQRSTUVWXYZ
Guess: L
Word:  - A L - A                  Lives: 6      Score: 15         BCDEFGHIJKMNOPQRSTUVWXYZ
Guess: N
Word:  - A L - A                  Lives: 5      Score: 15         BCDEFGHIJKMOPQRSTUVWXYZ
Guess: R
Word:  - A L - A                  Lives: 4      Score: 15         BCDEFGHIJKMOPQSTUVWXYZ
Guess: M
Word:  M A L - A                  Lives: 4      Score: 25         BCDEFGHIJKOPQSTUVWXYZ
Guess: A
You entered the same letter before.
Word:  M A L - A                  Lives: 4      Score: 25         BCDEFGHIJKOPQSTUVWXYZ
Guess: T
Word:  M A L T A                  Lives: 4      Score: 35         BCDEFGHIJKOPQSUVWXYZ

You win !!

High Score Table
Ali 50
Cem 40
Can 35
Deniz 35
...
Ece 5
```