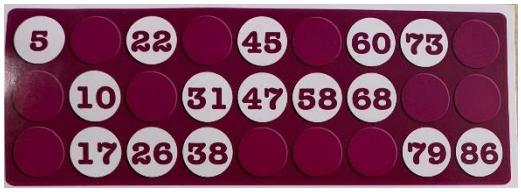
BLM19202E/ BLM22212E / SEN22212E Data Structures Project #1
Multi-Player Tombola (Bingo) Game with Multi Linked List
(March 31, 2024, 23:59)

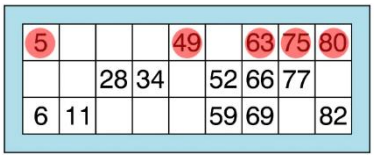
In this project, you are expected to develop a program in Java / C programming language that imitates the multi-player game known as "Tombola", using linked list data structure. Tombola is a popular game of chance played with numbered cards. The game involves randomly selecting numbers and players marking off those numbers on their cards if they match the ones called. You can find detailed information on the game at this link: https://themagictoyshop.co.uk/blogs/children-toys/how-to-play-bingo-in-the-uk-a-beginners-guide

HOW TO PLAY THE GAME

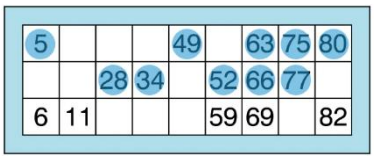
Each player has a tombola card, each consisting of 3 horizontal rows and 9 vertical columns, making a total of 27 boxes. Each card has 15 numbers arranged in three rows. There are 5 numbers in each horizontal row and the remaining 4 boxes are randomly blocked out. The first column has numbers from 1-9, the second from 10-19 and so on up to 90.



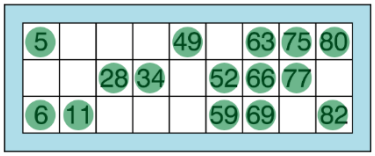
- At each step, a number is randomly selected without replacement (i.e. once a number is selected it cannot be selected again).
- Players should check their card to see if they can strike a number off.
- When a player marks five numbers in any row completely, the player must shout Bingo (birinci çinko)!



- The game goes to the next stage, in which the players must mark 10 numbers on any two lines. Then, the player must shout Bingo (ikinci çinko)!.

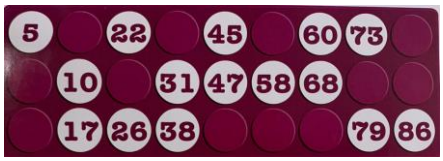


- The most thrilling part of the game is when the full house is played. The goal of all players is to mark 15 numbers out of 90 and to shout Bingo (Tombala!).



An example scenario is given below. At the end of each step, the current version of the cards should be printed on the screen. Let's say we have two players with the following cards:

User 1



User 2

Let the randomly selected numbers be in this order: 10, 4, 34, 58, 31, 45, 68, 60, 47, 80. The cards will be

User 1 says Bingo (birinci çinko) at that stage. Let the randomly selected numbers be in this order: 5, 26, 22, 38, 73, 77. The cards will be :

User 1 says Bingo (ikinci çinko!). Let the randomly selected numbers be in this order: 51, 17. The cards will be :

User 2 says Bingo (birinci çinko). Let the randomly selected numbers be in this order: 86, 79. The cards will be :

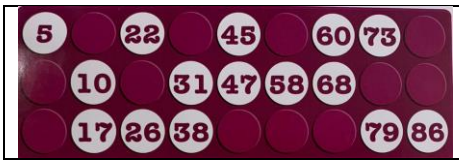
User 1 says Bingo (Tombala!).

APPLICATION DETAILS

The game is played with two players. Each player has a tombola card with 15 numbers, each of which is arranged in three rows. There are 5 numbers in each horizontal row and the remaining 4 boxes are randomly blocked out.

A linked list should be used for each card. Each node in the linked list will correspond to a box on the card. You ***should not store the blocked boxes*** on the card. You ***must use a multi-linked list*** which is a special type of list that contains one or more logical key sequences. You must reach all nodes of the multi-linked list via the **head** pointer. You can design your own node class with multiple pointers. You can move in one direction (like single linked list) or two (like doubly linked list) in the multi linked list.

In your main method, you should create a matrix that represents a card. The matrix (card) is randomly generated with respect to the card design, i.e. the first column has numbers from 1-9, the second from 10-19 and so on up to 90. You will need to create as many matrices as there are players in the game. At the end of each step, the current version of the cards should be printed on the screen.




```
int[][] card1 = {{5, -1, 22, -1, 45,  -1, 60, 73, -1},
                 {-1, 10, -1, 31, 47, 58, 68, -1, -1},
                 {-1, 17, 26, 38, -1, -1, -1, 79, 86}};
```

A random number is drawn at each step; therefore, you must generate a random permutation.

```
int[] randomPermutation = generatePermutation(90);
```

We check your program by changing these matrices and the random permutation.

All operations will be done using linked lists. Please don't use arrays instead, **the projects prepared with array or other data structures (ArrayList, etc.) will not be evaluated**.

You must write a report including your design and implementation details. You must draw your multi-linked list example in a figure. The report also includes the output of your program.

GRADING:

1. Multi linked list class(es) (10 points)
2. Game (75 points)
    a. Generating random permutation (5 points)
    b. Generating card for each user (15 points)
    c. Checking the status of card (birinci çinko, ikinci çinko, tombala) (45 points)
    d. Print the current version of the cards (10 points)
3. Report (15 points)

Bonus (15 points) – GUI: Game can be played in Java Frame; application interface will be prepared properly. You must design and implement your own GUI.

Notes:
By the due date, please submit the source code of your program, on the submission on LMS. And please name your files as follows: "*yournamesurnameXXX*.java", for example "BernaKirazMainClass.java".

Note that projects submitted after the project's due date will not be accepted and evaluated. Please keep this in mind and **promptly start working on your projects**!

You are required to exhibit an *individual effort* on this project. Similarity test is applied with all projects. Any potential violation of this rule will lead everyone involved to failing all projects.
Good luck ☺