CS 101 - Algorithms & Programming I

Fall 2022 - Lab 5

Due: Week of November 7, 2022

*Remember the **honor code** for your programming assignments.*
*For all labs, your solutions must conform to the CS101 style **guidelines**!*
All data and results should be stored in variables (or constants where appropriate) with meaningful names.

The objective of this lab is to learn how to use for and do-while loop to implement automated repetition. Remember that analyzing your problems and designing them on a piece of paper *before* starting implementation/coding is always a best practice.

In this particular lab, **only use the for or do-while loops**, do *not* use the while loop.

0. Setup Workspace

Start VSC and open the previously created workspace named `labs_ws`. Now, under the `labs` folder, create a new folder named `lab5`.

In this lab, you are to have two Java classes/files (under `labs/lab5` folder) as described below. A third Java file containing the revision should go under this folder as well. We expect you to submit a total of 3 files including the revision, **without compressing** them. Do *not* upload other/previous lab solutions in your submission. The user inputs in the sample runs are shown with blue color.

1. Falling Stars

Create a new/empty file of your own under the `lab5` folder named `Lab05_Q1.java` with a class with the same name. One fine day, you, a great astronomer, are observing the sky and witnessing that the stars are falling! You want to simulate this on your computer so you decide to write a program. In this program, you will get an input **string** composed of digits. Each digit within this string is actually the number of stars falling from the sky vertically. Your task is to print a "blueprint" of the falling stars using the asterisk symbol (*) and **for** loops. Do not use while or do-while loops in this part. You may assume that the input string is valid, consisting of only digits.

Sample run 1:

```
Please enter 'falling stars' as string: 2416
Falling stars:
2416
****
** *
 * *
 * *
   *
   *
```

Sample run 2:

```
Please enter 'falling stars' as string: 4858948200569652121
Falling stars:
4858948200569652121
********  *********
********  ****** *
*******    *****
*******    *****
 **** *    *****
 * ** *     ***
 * ** *      *
 * ** *      *
    *        *
```

2. Papaz Kaçtı (Card Game, similar with the 'Old Maid' game)

Create a new/empty file of your own under the `lab5` folder named `Lab05_Q2.java` with a class with the same name. Your program will be a simplified version of the popular card game "Papaz Kaçtı" (Similar with Old Maid). Even if you are familiar with the game, you should still read the instructions thoroughly, as ***there are quite a bit of simplifications and changes from the original game.*** A couple of sample runs of the game are added at the end of the explanations. Be careful about your outputs. They should be aligned with the given examples.

Remark1: You are ***not*** allowed to use any data type or class we have not yet learnt about (e.g. from the Collection framework such as Lists, Arrays and ArrayLists).

Our version of 'Papaz Kaçti' is a 2-player card game, the Player-1 and 2. We can see both players' hands. The card game is played with a standard 52-card deck with one of the Kings taken out so that only 51 cards remain. Details of the card deck are given in Section 2.1. The 51-cards are dealt out until all the cards have been dealt. It is okay that one player may have more cards than the other. The objective of the game is to avoid holding the last King.

Game play:
The game play starts when the deck is dealt. Each player looks at the cards dealt and removes any pairs from their own hands. Let's say Player-1 starts by drawing a random card from Player-2. If Player-1 has a pair with the drawn card, Player-1 should remove that pair from the hand. Then Player-2 draws a random card from Player-1 and again if the drawn card creates a pair, the matching cards should be removed from the hand. If the card drawn does not create a pair, the card stays in the player's hand.

When a player pairs up all the cards in their hand, that player will out the game. Since there are 2 players, this means that the other player holds the odd King in his hand. At the end, when all the pairs are matched and removed, the player having the odd King in the hand will lose the game.

Remark2: When the 51-card are dealt out, one player will get one more card than the other. In this case, *the player with less cards at initial card dealing should make the first move in the game*, which means that player with the less cards should draw a random card from the other player, alternating in this manner.

The game can be summarized with the following steps, each of which will be detailed in the following sections:

1. Deck creation & initial card drawn.
    a. Create deck and remove one King
    b. Deal all the cards to *two players*
    c. Remove all pairs from all two players' hands
2. Game Rounds
    a. If Player-1's turn, Player-1 draws a random card from Player-2
        i. Remove one randomly selected card from Player-2's hand and add that card to Player-1's hand.
        ii. Check for a new pair in Player-1's hand and remove if any.
    b. If Player-2's turn, Player-2 draws a random card from Player-1
        i. Remove one randomly selected card from Player-1's hand and add that card to Player-2's hand.
        ii. Check for a new pair in Player-2's hand and remove if any.

2.1. Deck creation & initial card drawn

- First, construct a **string** that will be used in place of a card deck. A standard card deck is composed of 13 ranks (numbers through 1 to 10, Jack, Queen and King) in 4 suits (categories like clubs, diamonds, etc.), for a total of 52 cards. For simplicity, our deck includes positive digits (1-9) and T,J,Q,K to represent 10, Jack, Queen and King, respectively. So T is for 10, J is for Jack, Q is for Queen and K is for King. We don't care about the suit of the card but there will be four of each card except one King. All in all, we are left with a 51-card deck where each digit/card is included 4 times except King. Use a **for** loop to create a string corresponding to a deck (except the one King) and print it:

```
Starting the game with the following deck:
111122223333444455556666777788889999TTTTJJJJQQQQKKK
```

- Before game rounds start, we deal ***all*** the cards to two players randomly. So, for each player, randomly select cards from the deck and add them to the player's hand which is in String format.

```
Initial cards are dealing...
Player-1 hand: KT195249KT5361J572TJQK8378
Player-2 hand: T12243434656 7Q88991JJQ7Q
```

Tip: To randomly select a card you can use the following code (assuming there are 5 cards in the deck):

```
import java.util.Random; // Similar to what we do with the Scanner
...
...
Random rand = new Random(); // Again, very similar to Scanner
int randomChoice = rand.nextInt(5); // get a random number in range [0,5)
```

- Remove all pairs from both players' hands. To do this, you need to find *duplicates* in Strings (players' hands) and remove them from the hands. Remember that if there are three cards with the same number (e.g. TTT) in one hand, you should remove only two of them.

```
Pairs are removed:
Player-1 hand: 456TQK
Player-2 hand: 456TQ
```

2.2. Game Round

- After our initialization is completed, we can now start the core game loop. Since in this example Player-1 has more cards (26 cards) when initial cards were dealt, Player-2 draws first (Remember: *the player with less card at initial card dealing should make the first move in the game*)

```
Start Drawing...
Player-2 is drawing the card: 3 from Player-1...
      Player-1 hand: 589
      Player-2 hand: 3589K3
```

- Now, Player-2 needs to pair 3s after the drawing card:3

```
Pairs are removed:
Player-1 hand: 589
Player-2 hand: 589K
```

- Now it is time for Player-1 to draw a card from Player-2

```
Start Drawing...
Player-1 is drawing the card: 8 from Player-2...
      Player-1 hand: 5898
      Player-2 hand: 59K
```

- Again the pairs (8s in this case) are removed from Player-1's hand

```
Pairs are removed:
Player-1 hand: 59
Player-2 hand: 59K
```

- Game continues with Player-2's round and removing pairs

```
Start Drawing...
Player-2 is drawing the card: 9 from Player-1...
      Player-1 hand: 5
      Player-2 hand: 59K9

Pairs are removed:
Player-1 hand: 5
Player-2 hand: 5K
```

- We are close to the end. Now it is Player-1's turn. Since cards are randomly drawn, Player-1 can draw 5 and win the game, or draw K. In the former case, the game should finish and you need to announce the winner as Player-1. In the latter case, however, the game should continue.

5 is drawn:

```
Start Drawing...
Player-1 is drawing the card: 5 from Player-2...
       Player-1 hand: 55
       Player-2 hand: K

Pairs are removed:
Player-1 hand:
Player-2 hand: K

Game Over!
Player-1 Wins!
       Player-1 hand:
       Player-2 hand: K
```

K is drawn:

```
Start Drawing...
Player-1 is drawing the card: K from Player-2...
       Player-1 hand: 5K
       Player-2 hand: 5

Pairs are removed:
Player-1 hand: 5K
Player-2 hand: 5
```

A couple of additional sample runs for the whole game are presented below.

Run 1

```
Starting the game with the following deck:
1111222233334444555566667777888899997TTTTJJJJQQQQKKK

Initial cards are dealing...
Player-1 hand: Q7T711K819561J9432J6Q6943K
Player-2 hand: 2T2T2334457565788898JTJKQQ

Pairs are removed:
Player-1 hand: 25689T
Player-2 hand: 25689TK

Start Drawing...
Player-2 is drawing the card: 6 from Player-1...
       Player-1 hand: 2589T
       Player-2 hand: 25689TK6

Pairs are removed:
Player-1 hand: 2589T
Player-2 hand: 2589TK

Start Drawing...
Player-1 is drawing the card: 2 from Player-2...
```

```
      Player-1 hand: 2589T2
      Player-2 hand: 589TK

Pairs are removed:
Player-1 hand: 589T
Player-2 hand: 589TK

Start Drawing...
Player-2 is drawing the card: T from Player-1...
      Player-1 hand: 589
      Player-2 hand: 589TKT

Pairs are removed:
Player-1 hand: 589
Player-2 hand: 589K

Start Drawing...
Player-1 is drawing the card: K from Player-2...
      Player-1 hand: 589K
      Player-2 hand: 589

Pairs are removed:
Player-1 hand: 589K
Player-2 hand: 589

Start Drawing...
Player-2 is drawing the card: K from Player-1...
      Player-1 hand: 589
      Player-2 hand: 589K

Pairs are removed:
Player-1 hand: 589
Player-2 hand: 589K

Start Drawing...
Player-1 is drawing the card: 5 from Player-2...
      Player-1 hand: 5895
      Player-2 hand: 89K

Pairs are removed:
Player-1 hand: 89
Player-2 hand: 89K

Start Drawing...
Player-2 is drawing the card: 9 from Player-1...
      Player-1 hand: 8
      Player-2 hand: 89K9

Pairs are removed:
Player-1 hand: 8
Player-2 hand: 8K

Start Drawing...
Player-1 is drawing the card: 8 from Player-2...
```

```
      Player-1 hand: 88
      Player-2 hand: K

Pairs are removed:
Player-1 hand:
Player-2 hand: K

Game Over!
Player-1 Wins!
      Player-1 hand:
      Player-2 hand: K
```

Run 2

```
Starting the game with the following deck:
1111222233334444555566667777888899999TTTTJJJJQQQQKKK

Initial cards are dealing...
Player-1 hand: 67KK7724242T536Q1JJ8697T81
Player-2 hand: 131233TT44555698Q899JJKQQ

Pairs are removed:
Player-1 hand: 23569Q
Player-2 hand: 23569QK

Start Drawing...
Player-2 is drawing the card: 6 from Player-1...
      Player-1 hand: 2359Q
      Player-2 hand: 23569QK6

Pairs are removed:
Player-1 hand: 2359Q
Player-2 hand: 2359QK

Start Drawing...
Player-1 is drawing the card: K from Player-2...
      Player-1 hand: 2359QK
      Player-2 hand: 2359Q

Pairs are removed:
Player-1 hand: 2359QK
Player-2 hand: 2359Q

Start Drawing...
Player-2 is drawing the card: 2 from Player-1...
      Player-1 hand: 359QK
      Player-2 hand: 2359Q2

Pairs are removed:
Player-1 hand: 359QK
Player-2 hand: 359Q

Start Drawing...
```

```
Player-1 is drawing the card: 9 from Player-2...
      Player-1 hand: 359QK9
      Player-2 hand: 35Q

Pairs are removed:
Player-1 hand: 35QK
Player-2 hand: 35Q

Start Drawing...
Player-2 is drawing the card: K from Player-1...
      Player-1 hand: 35Q
      Player-2 hand: 35QK

Pairs are removed:
Player-1 hand: 35Q
Player-2 hand: 35QK

Start Drawing...
Player-1 is drawing the card: K from Player-2...
      Player-1 hand: 35QK
      Player-2 hand: 35Q

Pairs are removed:
Player-1 hand: 35QK
Player-2 hand: 35Q

Start Drawing...
Player-2 is drawing the card: 5 from Player-1...
      Player-1 hand: 3QK
      Player-2 hand: 35Q5

Pairs are removed:
Player-1 hand: 3QK
Player-2 hand: 3Q

Start Drawing...
Player-1 is drawing the card: Q from Player-2...
      Player-1 hand: 3QKQ
      Player-2 hand: 3

Pairs are removed:
Player-1 hand: 3K
Player-2 hand: 3

Start Drawing...
Player-2 is drawing the card: K from Player-1...
      Player-1 hand: 3
      Player-2 hand: 3K

Pairs are removed:
Player-1 hand: 3
Player-2 hand: 3K

Start Drawing...
```

```
Player-1 is drawing the card: K from Player-2...
      Player-1 hand: 3K
      Player-2 hand: 3

Pairs are removed:
Player-1 hand: 3K
Player-2 hand: 3

Start Drawing...
Player-2 is drawing the card: 3 from Player-1...
      Player-1 hand: K
      Player-2 hand: 33

Pairs are removed:
Player-1 hand: K
Player-2 hand:

Game Over!
Player-2 Wins!
      Player-1 hand: K
      Player-2 hand:
```