TCP1201 Objected-Oriented Programming and Data Structures
Assignment Part 1

Trimester 2, Session 2022/2023
Faculty of Computing and Informatics
Multimedia University

Part 1 Submission Deadline: 21 May 2023 (Sunday), 11:59pm
Part 2 Submission Deadline: 18 June 2023 (Sunday), 11:59pm

General Information

A. This 2-part assignment contributes **40%** of the total course mark.
B. Part 1 contributes 10% while Part 2 contributes 30%.
C. Part 2 is an extension of Part 1.
D. Part 1 presentation is a video presentation. Part 2 submission is a live presentation. All members shall present in both presentations.
E. ZERO mark for late submission.
F. STRICTLY NO COPYING from other students in other groups or from any other sources (Internet, books, etc.). ZERO mark to students who plagiarize AND to students who share their code. For this assignment, plagiarism means the following:
    1. Turning in a work that, from the examiner's point of view, you do not sufficiently understand.
    2. Turning in someone else's work (whether partly or fully) as your own.
    3. Any means of cheating.

Grouping

A. Max 4 members per group from the same tutorial section.
B. Register your group with your lab lecturer.
C. To detect a sleeping member early, form a group and start doing the assignment as early as possible. Have meetings regularly, e.g., twice a week, and get every member to present his/her work to the group during the meeting. Frequently not showing up or having nothing to present is a sign of sleeping member.
D. Each group member may be given different marks according to their contributions.

Go Boom Game

A. Watch How To Play Go Boom YouTube to understand the game.
B. Notice that the **start of the game** in this assignment is **different** from the YouTube video above.
C. Your task is to develop an object-oriented Java program for the Go Boom game.
D. Part 1 features (console output):
    1) All cards should be faced up to facilitate checking.
    2) Start a new game with randomized 52 cards.
    3) The first card in the deck is the first lead card and is placed at the center.
    4) The first lead card determines the first player:
        - A, 5, 9, K for Player1
        - 2, 6, 10 for Player2
        - 3, 7, J for Player3
        - 4, 8, Q for Player4
    5) Deal 7 cards to each of the 4 players.
    6) All players must follow the suit or rank of the lead card.
    7) The highest-rank card with the same suit as the lead card wins the trick.
    8) The winner of a trick leads the next card.
E. Part 2 features (console & GUI)
    1) If a player cannot follow suit or rank, the player must draw from the deck until a card can be played.
    2) When the remaining deck is exhausted and the player cannot play, the player skips (does not play) the trick.

3) Finish a round of game correctly. Display the score of each player. (No need to implement stopping the game at 100 points because there won't be sufficient time for that during presentation.)
4) Support GUI mode. The GUI can be in JavaFX, Swing, Spring, or Android.
5) Keep the console output to facilitate checking. The contents in both console output and the GUI must tally.

F. Sample output

At the beginning of a game. The first lead card s7 is placed at the center. Player3 is the first player because of s7. Player3 plays s3.

c = club
d = diamond
h = heart
s = spade

```
Trick #1
Player1: [c9, h5, dQ, d6, c7, s8, hQ]
Player2: [h6, sJ, d2, hX, d5, cA, hA]
Player3: [d9, dA, h7, d8, s3, sX, h2]
Player4: [h8, sA, sK, hK, c6, s4, h9]
Center : [s7]
Deck   : [c2, s5, s2, cQ, c5, dK, cJ, cX, hJ, c4, c8, dX, s6, c3, s9, d7, dJ, d3, cK, sQ, h4,
h3, d4]
Score: Player1 = 0 | Player2 = 0 | Player3 = 0 | Player4 = 0
Turn : Player3
> s3

Trick #1
Player1: [c9, h5, dQ, d6, c7, s8, hQ]
Player2: [h6, sJ, d2, hX, d5, cA, hA]
Player3: [d9, dA, h7, d8, sX, h2]
Player4: [h8, sA, sK, hK, c6, s4, h9]
Center : [s7, s3]
Deck   : [c2, s5, s2, cQ, c5, dK, cJ, cX, hJ, c4, c8, dX, s6, c3, s9, d7, dJ, d3, cK, sQ, h4,
h3, d4]
Score: Player1 = 0 | Player2 = 0 | Player3 = 0 | Player4 = 0
Turn : Player4
> s4

Trick #1
Player1: [c9, h5, dQ, d6, c7, s8, hQ]
Player2: [h6, sJ, d2, hX, d5, cA, hA]
Player3: [d9, dA, h7, d8, sX, h2]
Player4: [h8, sA, sK, hK, c6, h9]
Center : [s7, s3, s4]
Deck   : [c2, s5, s2, cQ, c5, dK, cJ, cX, hJ, c4, c8, dX, s6, c3, s9, d7, dJ, d3, cK, sQ, h4,
h3, d4]
Score: Player1 = 0 | Player2 = 0 | Player3 = 0 | Player4 = 0
Turn : Player1
> c7

Trick #1
Player1: [c9, h5, dQ, d6, s8, hQ]
Player2: [h6, sJ, d2, hX, d5, cA, hA]
Player3: [d9, dA, h7, d8, sX, h2]
Player4: [h8, sA, sK, hK, c6, h9]
Center : [s7, s3, s4, c7]
Deck   : [c2, s5, s2, cQ, c5, dK, cJ, cX, hJ, c4, c8, dX, s6, c3, s9, d7, dJ, d3, cK, sQ, h4,
h3, d4]
Score: Player1 = 0 | Player2 = 0 | Player3 = 0 | Player4 = 0
Turn : Player2
> sJ
```

```
*** Player2 wins Trick #1 ***

Trick #2
Player1: [c9, h5, dQ, d6, s8, hQ]
Player2: [h6, d2, hX, d5, cA, hA]
Player3: [d9, dA, h7, d8, sX, h2]
Player4: [h8, sA, sK, hK, c6, h9]
Center : []
Deck   : [c2, s5, s2, cQ, c5, dK, cJ, cX, hJ, c4, c8, dX, s6, c3, s9, d7, dJ, d3, cK, sQ, h4,
h3, d4]
Score: Player1 = 0 | Player2 = 0 | Player3 = 0 | Player4 = 0
Turn : Player2
>
```

G. Commands to the Program

The game shall support the following commands:

1. s – Start a new game.
2. x – Exit the game.
3. d – Draw cards from deck until a playable card is obtained. If the deck is empty, skip to the next player.
4. *card* – a card played by the current player.

Part 1 Submission Format

Your group shall submit the works to the GitHub under your MMU GitHub Education account. Make the repo **private** so that non-members cannot access your code. Add your lab lecturer as a **collaborator**.

1. Submit the following items to your group's GitHub repo. One group makes one submission.
    a. **src** folder – contains your **source code and any data files** (*.java, etc.)
    b. **PART1.md** – Fill in the **member contribution**, **feature completion**, and a **link to video presentation** (not more than 10 minutes).
2. Paste your group's GitHub link into the Google Classroom for Assignment Part 1.

Part 1 Marking Rubric

Item	Mark
1 Part 1 Features (8 marks) 1 mark for a fully working feature. 0.5 mark for a partially working feature.	
2 Video Presentation (2 marks) 2 marks for presenting all 8 features excellently. 1 mark for presenting some features adequately.	
Part 1 Total (Zero mark for no presentation, late submission, or plagiarism)	