TCP1201 Objected-Oriented Programming and Data Structures

Assignment

Trimester 1, Session 2017/2018

Faculty of Computing and Informatics

Multimedia University

**DUE DATE: 17 Sept 2017 (Sunday), 11:59pm**

**Outline**

This assignment contributes 20% of total subject mark. The assignment consists of only one question. Every student does and submits the assignment individually (upload to MMLS). Interview will be conducted after the due date. You are strongly advised to submit the assignment in time even though some features are incomplete (make sure the program is compile-able and some features work). For late submission, code that cannot be compiled or executed, zero mark will be given. No extension of dateline will be entertained so that the lecturer will have sufficient time to assess and release the mark before the final exam. Exact interview date and time will be scheduled after the due date.

Make sure your program code can be compiled and executed under JDK 8. JDK 8 is downloadable at <http://www.oracle.com/technetwork/java/javase/downloads/jdk8-downloads-2133151.html>

**Problem Statement**

You are developing a card game for 4 players. The game requires only one deck of unique cards as shown below:

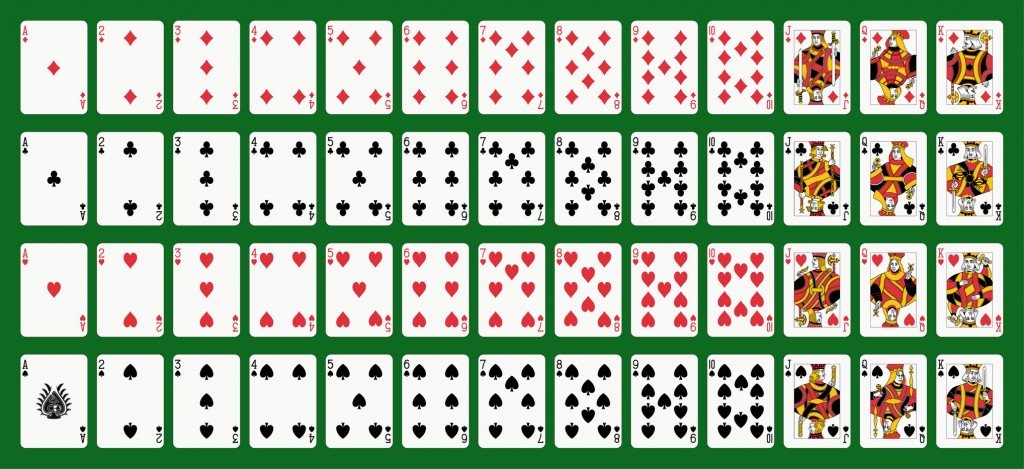


Photo Credit: Mannaggia / Fotolia

The deck of 52 card has 4 suits from top to bottom: diamonds (D), clubs (C), hearts (H), and spades (S). Each card has a point as shown in the table below.

|  |  |
| --- | --- |
| Face | Point |
| Ace/A | 1 |
| 2 | 2 |
| 3 | 3 |
| 4 | 4 |
| 5 | 5 |
| 6 | 6 |
| 7 | 7 |
| 8 | 8 |
| 9 | 9 |
| 10 | 0 |
| Jack/J | 0 |
| Queen/Q | 0 |
| King/K | 0 |

The 52 cards are shuffled at the beginning of a game. A game has many rounds. In each round, each player is given 3 cards based on a sequence of one card per player. The sum of cards for each player is calculated. The player(s) with the highest rightmost digit in the sum wins that particular round. The game will end when there are insufficient cards to continue to next round. The player with the highest win count wins the game.

Sample run:

Win count: P1 = 0 P2 = 0 P3 = 0 P4 = 0

Available cards: C5 S6 SK DA CK D4 H7 C4 CQ SA CA D10 H6 DQ D9 H5 H3 DJ D8 S7 D5 S3 D3 H2 H8 C8 H9 HQ SQ CJ C10 C2 H4 HA D2 C9 HK D6 SJ S10 DK C6 S9 S4 C7 HJ H10 S2 S8 D7 S5 C3

Used cards:

C:

D:

H:

S:

Press ENTER to deal cards...

ROUND 1:

P1 cards: C3 S2 S4 | Sum = 9 | Winner

P2 cards: S5 H10 S9 | Sum = 4

P3 cards: D7 HJ C6 | Sum = 3

P4 cards: S8 C7 DK | Sum = 5

Win count: P1 = 1 P2 = 0 P3 = 0 P4 = 0

Available cards: C5 S6 SK DA CK D4 H7 C4 CQ SA CA D10 H6 DQ D9 H5 H3 DJ D8 S7 D5 S3 D3 H2 H8 C8 H9 HQ SQ CJ C10 C2 H4 HA D2 C9 HK D6 SJ S10

Used cards:

C: 3 6 7

D: 7 K

H: 10 J

S: 2 4 5 9 8

Press ENTER to deal cards...

ROUND 2:

P1 cards: S10 C9 C2 | Sum = 1

P2 cards: SJ D2 C10 | Sum = 2

P3 cards: D6 HA CJ | Sum = 7 | Winner

P4 cards: HK H4 SQ | Sum = 4

Win count: P1 = 1 P2 = 0 P3 = 1 P4 = 0

Available cards: C5 S6 SK DA CK D4 H7 C4 CQ SA CA D10 H6 DQ D9 H5 H3 DJ D8 S7 D5 S3 D3 H2 H8 C8 H9 HQ

Used cards:

C: 3 6 7 9 2 10 J

D: 7 K 2 6

H: 10 J A K 4

S: 2 4 5 9 8 10 J Q

Press ENTER to deal cards...

ROUND 3:

P1 cards: HQ H2 S7 | Sum = 9 | Winner

P2 cards: H9 D3 D8 | Sum = 0

P3 cards: C8 S3 DJ | Sum = 1

P4 cards: H8 D5 H3 | Sum = 6

Win count: P1 = 2 P2 = 0 P3 = 1 P4 = 0

Available cards: C5 S6 SK DA CK D4 H7 C4 CQ SA CA D10 H6 DQ D9 H5

Used cards:

C: 3 6 7 9 2 10 J 8

D: 7 K 2 6 3 8 J 5

H: 10 J A K 4 Q 2 9 8 3

S: 2 4 5 9 8 10 J Q 7 3

Press ENTER to deal cards...

ROUND 4:

P1 cards: H5 D10 C4 | Sum = 9 | Winner

P2 cards: D9 CA H7 | Sum = 7

P3 cards: DQ SA D4 | Sum = 5

P4 cards: H6 CQ CK | Sum = 6

Win count: P1 = 3 P2 = 0 P3 = 1 P4 = 0

Available cards: C5 S6 SK DA

Used cards:

C: 3 6 7 9 2 10 J 8 4 A Q K

D: 7 K 2 6 3 8 J 5 10 9 Q 4

H: 10 J A K 4 Q 2 9 8 3 5 7 6

S: 2 4 5 9 8 10 J Q 7 3 A

P1 wins this game! Congrats!

Press Q to quit the game. Press N to start a new game...

**Submission Format**

1. A zip file named "*TT0X\_StudentName*.zip" where TT0*X* is your lab section and *StudentName* is your name. The zip should contain your **source code files** (\*.java) and **Javadoc documentation**.
2. Upload the zip to MMLS.

**Feature Sheet & Evaluation Criteria**

|  |  |
| --- | --- |
| **Criteria** | **Item** |
| 1. Program Design (8 marks) | 1.1. Style (indentation, documentation using javadoc, and naming) & Modularity (classes and methods) [2m] |
| 1.2. Association, Aggregation, and/or Composition [2m] |
| 1.3. Inheritance, Polymorphism, Abstract Class, and/or Interface [1m] |
| 1.4. Use adequate data structures (more than array and ArrayList) [3m] |
| 1. Program execution (12 marks. 0 if code unable to compile or run.) | 2.1. User friendliness (input & output sufficiently self-explain) [1m] |
| 2.2. Correctly deal cards to players [2m] |
| 2.3. Correct sum of cards for each player [2m] |
| 2.4. Correct win count for each player [2m] |
| 2.5. Correct history of used cards [2m] |
| 2.6. Can complete a game and start a new game [1m] |
| 2.7. Randomly shuffle 52 unique cards at the beginning of game [2m] |
| 1. Bonus (2 marks)   Total assignment mark is capped at 20. | For additional significant features only, such as GUI, or support also 3 or 5 players, etc. (Check with your lecturer for acceptable features) |
| 1. Interview (0 mark if fail to be present for interview. Mark will be deducted if unable to explain.) | 3.1. Fluency in using the program |
| 3.2. Ability to explain code |
| 1. Plagiarism, late submission | 0 mark for the whole assignment |