Assignment 1
War Card Game

CSIS 3475

War card game overview

- War (US) or Battle (UK) is a card game typically played by two players. It uses a standard 52 card deck in decreasing order: A K Q J 10 9 8 7 6 5 4 3 2.
- The objective of the game is to win all cards.
- The deck is divided evenly among the players, giving each a **cards to be played** deck. In unison, each player reveals the top card of their deck—this is a "battle"—and the player with the higher card takes both cards played and moves them to their **won** deck. Aces are high, and suits are ignored.
- If the two cards played are of equal value, then there is a "war". Both players place the next three cards from their **cards to be played** deck face down onto the **war** deck, and then another card face-up on the **war** deck. The owner of the higher face-up card wins the war and adds all **ten** cards on the table to their **won** deck. If the face-up cards are again equal then the battle repeats with another set of face-down/up cards. This repeats until one player's face-up card is higher than their opponent's.
- When there are no more cards in the **cards to be played** deck of cards, the **won** deck of cards is shuffled and then used as the **cards to be played** deck. Basically, the **won** deck of cards are moved to the **cards to be played** deck after shuffling.
- If a player runs out of cards during a battle or war, that player immediately loses and the game ends.
- In this variant of the game, note that each player has three stacks.
- For more information, see https://en.wikipedia.org/wiki/War_(card_game) and https://www.pagat.com/war/war.html
- A nice graphical simulation is at https://cardgames.io/war, but you may want to be careful with this site.

Assignment

- Download **Assignment 1.zip** and import it into an Eclipse workspace using the standard instructions.
- Copy your implementations of Stacks and Queues to the StackPackage and QueuePackage. Make sure they are tested.
  - Include ALL implementations, including Deque
- Complete the game in the WarGamePackage
  - Test thoroughly with Stacks and Queues.
- Submit the completed projects using the standard submission instructions
  - Export the projects to a zip archive named **Assignment 1 YourName.zip** where YourName must be your first name and last name.
  - You MUST use the submission instructions exactly or you will lose marks.
  - You MUST name the archive correctly or you will lose marks.
  - For example, for Michael Hrybyk
    - Assignment 1 MichaelHrybyk.zip
    - Note NO SPACES in YourName

WarGamePackage

- This package has all of the files needed to complete the program.
  - WarCardGame.java
    - main program and logic
    - creates main deck, shuffles it, and deals the cards initially to two players
    - loop logic for battles and wars
    - determines when game is over (loser declared)
  - DeckUsingStack.java and DeckUsingQueue.java
    - extends abstract class Deck implementing DeckInterface
    - DeckUsingStack uses an internal Stack
    - DeckUsingQueue uses an internal Queue
    - shuffle method
  - Player.java
    - contains the three decks (cardsToBePlayed, wonCards, and war)
    - has methods to add and remove cards
    - implements PlayerInterface.java
  - Card.java
    - a single card with a card type, suit, number, and rank
    - CardType.java – enum for a number or face card
    - Suit.java – enum for card suits

War Game Tasks

- Complete each task below in the project. Make sure you write code that tests each task thoroughly as you finish it.
- Task 1
  - Complete all Stack and Queue implementations in the StackPackage and QueuePackage.
  - In DeckUsingStack.java and DeckUsingQueue.java, comment out the implementation you want to use first. Others can be tested once you have the rest of the program done.
  - Study the Deck abstract class and DeckInterface. Study the shuffle method and understand how it works.
- Task 2
  - Complete the PlayerInterface methods in Player.java which uses three Decks.
  - Do not change the toString() method.
  - Comment out the Deck implementation to use – DeckUsingStack() or DeckUsingQueue().
- Task 3
  - Using Player and DeckUsing{Stack,Queue} classes, implement the game in WarCardGame.java according to the rules laid out in the overview.
    - Complete the createStandardDeck() and displayDeck() methods.
  - Test your code thoroughly. Output should correspond in format exactly to that contained in StackOutput.txt and QueueOutput.txt
    - although due to random nature of the game and shuffling, each game played will have a different number of rounds and cards won/played.
    - Note that cards that are played are pulled from the front for Queue, and from the back for Stack.

Grading

Item	Marks
Project properly named and submitted. All code properly formatted and commented using Javadoc standard.	.4
ALL StackPackage and QueuePackage classes completed and tested	1.4
Player.java completed and tested with Stacks and Queues	1.6
WarGame.java completed.	1.6
Total	5