

Add size() function to Deck

Add isEight() function to Card

Add setSuit(string) function to Card

Add determineWinner() function to Game

Game

Mocks: UI, Deck

1. Deals card to two players
 - a. Each player's hand size is 5
 - b. Hand::addToHand is called 10 times
 - c. Deck::getTopCard is called 10 times
 - d. Deck::size is called once
2. Deals card to four players
 - a. Each player's hand size is 5
 - b. Hand::addToHand is called 20 times
 - c. Deck::getTopCard is called 20 times
 - d. Deck::size is called once
3. Play an 8 and pick the suit
 - a. Hand::addToHand is called after Deck::shuffleCards
 - b. Deal cards
 - i. Deal 1 Ace of Hearts, 2 Two of Hearts, 1 Seven of Clubs, 1 Eight of Spades to one player
 - ii. Deal 1 Queen of Spades, 1 Jack of Spades, 1 Six of Hearts, 1 Queen of Diamonds, 1 Three of Diamonds to the other player
 - c. Deck::size() is called once and returns 42
 - d. Show the top card by using Deck::getTopCard
 - e. PlayerController::playCard()
 - i. Called once
 - f. TurnUI::getPlayedCard
 - i. Called once
 - g. Card::isEight()
 - i. Called once
 - ii. If true pick either Clubs, Spades, Diamonds, or Hearts and then call Card::setSuit()
 - h. Card::getSuit()
 - i. Called once
4. Determine winner and calculate score
 - a. Hand::addToHand is called after Deck::shuffleCards
 - b. Deal cards
 - i. Deal 1 Ace of Hearts, 2 Two of Hearts, 1 Seven of Clubs, 1 Eight of Spades to 1st player
 - ii. Deal 1 Queen of Spades, 1 Jack of Spades, 1 Six of Hearts, 1 Queen of Diamonds, 1 Three of Diamonds to 2nd player
 - iii. Deal 0 cards to 3rd player

- c. isGameOver()
 - i. Called once
- d. determineWinner()
 - i. Called once and prints the name of the winner
 - 1. Player::getName()
- e. Player::addToScore
 - i. Called once
 - ii. Score for first player should be 60
 - iii. Score for second player should be 39
 - iv. Third player is the winner since they have no cards in their hand
- f. EndGameUI::getQuit()
 - i. Called once
- g. Game::endGame()
 - i. Called once
- 5. Starting game
 - a. Game::startGame()
 - i. Called once
 - b. StartGameUI::getNumOfPlayers()
 - i. Set number of players to three
 - c. DeckFactory::makeDeck()

PlayerController

Mocks UI

- 1. Test PlayerController(player*)
 - a. Call Player->getHand
 - b. Player::getHand()
 - c. Call Player->getScore
 - d. Player::getScore()
 - e. Call Player->getName()
 - f. Player::getName()
 - g. Call Player->addToScore(x)
 - h. Player::addToScore(x)
- 2. SortHand()
 - Order cards by suit or rank
- 3. playCard()
 - Return pointer to card we want to play
- 4. receiveCard()
 - Take in card to use in hand

UI

- 1. Tested by system-level testing after implementation.

TurnUI

1. Tested by system-level testing after implementation.

StartGameUI

1. Tested by system-level testing after implementation.

EndGameUI

1. Tested by system-level testing after implementation.

HelpUI

1. Tested by system-level testing after implementation.

Player

1. Tested indirectly in other tests:
 - a. getHand()
 - b. getScore()
 - c. getName()
 - d. addToScore()
 - e. playCard()

CI

Mocks: Deck

1. playCard() draw
The ai is unable to play, will its algorithm make it draw?
2. playCard() play suit
Ai has matching suit will it play the corresponding card

Human

1. Tested by system-level testing after implementation.

Hand

1. Tested by system-level testing after implementation.

Card

1. Tested by system testing.

Deck

1. Shuffle cards and see if the cards go into a different order

2. `getTopCard` should return a card. `getTopCard` should also remove the card from the deck
3. `recieveCard` will add a card to the deck
4. Deck size will decrease as a card gets drawn

DeckFactory

1. Tested by system-level testing after implementation.