

CLASS: MainGame

| METHOD | # | DESCRIPTION | SAMPLE INPUT / STATUS | EXPECTED OUTPUT | ACTUAL OUTPUT | P/F |
|--|---|---|-------------------------------|--|---|-----|
| preliminaryStart() | 1 | Input number of players is outside valid input (2-3) | 4 | Invalid input | Invalid input | P |
| | | Number of players instantiated is equal to input number of players | 2 | <Upon enter to Phase1Game, show Player 1 and Player 2 turns before cycling back to Player 1> | <Program shows Player 1 turn, then Player 2 turn, before cycling back to Player 1> | P |
| | | Each player's career is pre-defined (will obviously change during phase 2) | N/A (Provided at program run) | <Each player's career is Athlete (String)> | <Each player's career is Athlete (String)> | P |
| generateDeckOfActionCards() | 2 | 20 Collect From Bank and 20 Pay The Bank action cards are instantiated | N/A (Provided at program run) | <20 Action cards with ID 0 and 20 Action cards with ID 1 is added into ArrayList<ActionCards>> | <20 Action cards with ID 0 and 20 Action cards with ID 1 is added into ArrayList<ActionCards>> | P |
| | | 5 Pay The Player and 5 Collect From Player action cards are instantiated | N/A (Provided at program run) | <5 Action cards with ID 2 and 5 Action cards with ID 3 is added into ArrayList<ActionCards>> | <5 Action cards with ID 2 and 5 Action cards with ID 3 is added into ArrayList<ActionCards>> | P |
| | | Cards generated are immediately shuffled before the start of the first game. | N/A (Provided at program run) | <Cards are shuffled and its order is unique every launch of the program> | <Cards are shuffled and its order is unique every launch of the program> | P |
| | | With -ac argument, user specifies at what index to start drawing an action card (before the first round starts) | 10 | <Action card starts drawing at index 9, upon drawing all cards reshuffle, start at index 49> | <Action card starts drawing at index 9, upon drawing all cards reshuffle, start at index 49> | P |
| displayActionCards(ArrayList<ActionCard> deck) | 3 | On program run, this method is called to show the generated shuffled action cards. | N/A (Provided at program run) | Action Cards generated (uses stack implementation): LAWSUIT, SETUP SCHOOL, BONUS PAYDAY, SETUP SCHOOL, SELL AN ITEM, WATCH A SHOW, SETUP SCHOOL, IT'S YOUR BIRTHDAY!, WATCH A SHOW, BUY AN ITEM!, WIN A COMPETITION, TAX REFUND, TAX REFUND, WRITE A BOOK, VISIT A PLACE, WIN A COMPETITION, HIKING, FILE A LAWSUIT!, SETUP SCHOOL, IT'S YOUR BIRTHDAY!, SETUP SCHOOL, HIKING, HIKING, SELL AN ITEM, WIN A | Action Cards generated (uses stack implementation): LAWSUIT, SETUP SCHOOL, BONUS PAYDAY, SETUP SCHOOL, SELL AN ITEM, WATCH A SHOW, SETUP SCHOOL, IT'S YOUR BIRTHDAY!, WATCH A SHOW, BUY AN ITEM!, WIN A COMPETITION, TAX REFUND, TAX REFUND, WRITE A BOOK, VISIT A PLACE, WIN A COMPETITION, HIKING, FILE A LAWSUIT!, SETUP SCHOOL, IT'S YOUR BIRTHDAY!, SETUP SCHOOL, HIKING, HIKING, SELL AN ITEM, WIN A COMPETITION, TAX | P |

| | | | | | | |
|----------------------|---|--|---|---|--|---|
| | | | | COMPETITION, TAX REFUND, WRITE A BOOK, WATCH A SHOW, TRAFFIC VIOLATION, BUY AN ITEM!, BUY AN ITEM!, LAWSUIT, HIKING, LAWSUIT, SETUP SCHOOL, WIN A COMPETITION, WRITE A BOOK, VISIT A PLACE, LAWSUIT, SETUP SCHOOL, SETUP SCHOOL, SELL AN ITEM, FILE A LAWSUIT!, BONUS PAYDAY, WATCH A SHOW, BUY AN ITEM!, HIKING, LAWSUIT, FILE A LAWSUIT!, TAX REFUND, | REFUND, WRITE A BOOK, WATCH A SHOW, TRAFFIC VIOLATION, BUY AN ITEM!, BUY AN ITEM!, LAWSUIT, HIKING, LAWSUIT, SETUP SCHOOL, WIN A COMPETITION, WRITE A BOOK, VISIT A PLACE, LAWSUIT, SETUP SCHOOL, SETUP SCHOOL, SELL AN ITEM, FILE A LAWSUIT!, BONUS PAYDAY, WATCH A SHOW, BUY AN ITEM!, HIKING, LAWSUIT, FILE A LAWSUIT!, TAX REFUND, | |
| phase1Game(int turn) | 4 | Round 1 turn, display Player 1 status (No card has been drawn yet) | N/A | PLAYER 1's turn! MONEY: 200000.0 LOAN: 0.0 | PLAYER 1's turn! MONEY: 200000.0 LOAN: 0.0 | P |
| | | At round 1, Player 1 drawn action card is WIN A COMPETITION (\$30000 deducted) | N/A | Action card drawn: WIN A COMPETITION Put your skills to the test! \$20000 deducted from Player 1 Updated current MONEY: 170000.0 | Action card drawn: WIN A COMPETITION Put your skills to the test! \$20000 deducted from Player 1 Updated current MONEY: 170000.0 | P |
| | | At end of each round, increment player turn | N/A (current turn is player 1) | <Proceed next turn for player 2> | <Proceed next turn for player 2> | P |
| | | At round 2, Player 2 draws action card LAWSUIT (\$30000 deducted to chosen player) to player 3 | 3 | PLAYER 2's turn! MONEY: 200000.0 LOAN: 0.0 Action card drawn: LAWSUIT Settle a case! Choose player number to pay: 3 Updated current MONEY: \$170000.0 ----- PLAYER 3's turn! MONEY: 230000.0 LOAN: 0.0 | PLAYER 2's turn! MONEY: 200000.0 LOAN: 0.0 Action card drawn: LAWSUIT Settle a case! Choose player number to pay: 3 Updated current MONEY: \$170000.0 ----- PLAYER 3's turn! MONEY: 230000.0 LOAN: 0.0 | P |
| | | Draw an action card, deck not yet empty | N/A (Automatic, top of the card is FILE A LAWSUIT!) | <Drawn card is FILE A LAWSUIT> | <Drawn card is FILE A LAWSUIT> | P |
| | | Drawn an action card, | N/A (Last card on the | DECK RAN OUT! Cards | DECK RAN OUT! Cards to | P |

| | | | | | | |
|------------|---|--|------------------------------|---|--|---|
| | | deck is about to become empty | <i>deck is SELL AN ITEM)</i> | to be automatically reshuffled at NEXT turn. | be automatically reshuffled at NEXT turn. | |
| | | Drawn action card deducts a player's money, player's money is not enough Drawn card is WIN A COMPETITION (deduct \$30000.00) and current player's balance is \$20000.00 | N/A (Handled by the program) | Warning! A loan added to current player. Make sure to pay it back. Updated current loan and interest total is now 25000.0 \$30000 deducted from Player 1 Updated current MONEY: 10000.0` | Warning! A loan added to current player. Make sure to pay it back. Updated current loan and interest total is now 25000.0 \$30000 deducted from Player 1 Updated current MONEY: 10000.0 | P |
| drawDeck() | 5 | Draw the top most card from the deck. Deck is not empty | N/A | <The top most card from the deck is drawn.> | <The top most card from the deck is drawn.> | P |
| | | Draw the top most card from the deck. Deck is empty. | N/A | DECK RAN OUT! Cards to be automatically reshuffled at NEXT turn. | DECK RAN OUT! Cards to be automatically reshuffled at NEXT turn. | P |

CLASS: Player

| METHOD | # | DESCRIPTION | SAMPLE INPUT | EXPECTED OUTPUT | ACTUAL OUTPUT | P/F |
|--------------------------------|---|---|--------------|---|---|-----|
| addMoneyBalance(double amount) | 3 | Adds amount to the current money balance of the player | 30000.00 | <30000.00 is added to the current balance of the player> | <30000.00 is added to the current balance of the player> | P |
| addLoan() | 4 | Player 1 currently has \$20000 balance, draws VISIT A PLACE (deduct \$30000) | N/A | Warning! A loan added to a player. Make sure to pay it back. Updated loan and interest for a player total is now 25000.0 \$30000 deducted from Player 1 Updated current MONEY: 10000.0 | Warning! A loan added to a player. Make sure to pay it back. Updated loan and interest for a player total is now 25000.0 \$30000 deducted from Player 1 Updated current MONEY: 10000.0 | P |
| | | Player 1 needs to make more than one loan to pay the other players. Drawn card is CHRISTMAS BONUS (Pay all players \$30000) | N/A | Warning! A loan added to a player. Make sure to pay it back. Updated loan and interest for a player total is now 25000.0 PAID Player 2 \$30000.00 Warning! A loan added to a player. Make sure to pay it back. Updated loan and interest for a player total | Warning! A loan added to a player. Make sure to pay it back. Updated loan and interest for a player total is now 25000.0 PAID Player 2 \$30000.00 Warning! A loan added to a player. Make sure to pay it back. Updated loan and interest for a player total | P |

| | | | | | | |
|--------------------------------------|---|---|----------|---|--|---|
| | | | | is now 50000.0 PAID Player 3 \$30000.00 Updated current MONEY: \$0.0 | 50000.0 PAID Player 3 \$30000.00 Updated current MONEY: \$0.0 | |
| reduceMoneyBalance(double amount) | 5 | Subtracts amount to the current money balance of the player. If the amount goes lower than 0, addLoan() will be called. | 40000.00 | <40000.00 is subtracted from the balance of the player> | <40000.00 is subtracted from the balance of the player> | P |
| setCareer() | 6 | Career is predefined as "Athlete" | N/A | Player career: Athlete | Player career: Athlete | P |

CLASS: ActionCard

| METHOD | # | DESCRIPTION | SAMPLE INPUT | EXPECTED OUTPUT | ACTUAL OUTPUT | P/F |
|---|---|---|---------------------------------|--|---|-----|
| generateSubID() | 7 | Collect From Bank cards get subID from 0 to 4 | N/A | <Generated Collect From Bank cards get subID ranges from 0 to 4> | <Generated Collect From Bank cards get subID ranges from 0 to 4> | P |
| | | Pay The Bank cards get subID from 0 - 5 | N/A | <Generated Pay The Bankcards get subID ranges from 0 to 5> | <Generated Pay The Bankcards get subID ranges from 0 to 5> | P |
| | | Pay The Player cards get subID from 0-1 | N/A | <Generated Pay The Player get subID ranges from 0 to 1> | <Generated Pay The Player get subID ranges from 0 to 1> | P |
| assignDescriptions() | 8 | Assign appropriate description for a particular Collect From Bank card | MainID : 0 subID : 0 | TypeOfCard = TAX REFUND Description: Get your money back from the taxpayers! toDoAction: Collect \$30000 From The Bank | TypeOfCard = TAX REFUND Description: Get your money back from the taxpayers! | P |
| | | Assign appropriate description for a particular Pay The Bank card | MainID : 1 subID : 1 | TypeOfCard: VISIT A PLACE Description: Go spend some good time in a good place! toDoAction: Pay \$30000 To The Bank | TypeOfCard: VISIT A PLACE Description: Go spend some good time in a good place! | P |
| | | Assign appropriate description for a particular Pay The Player card | MainID : 2 subID : 0 | TypeOfCard: LAWSUIT Description: Settle a case toDoAction: Choose A Player, then Pay \$30000 | TypeOfCard: LAWSUIT Description: Settle a case toDoAction: Choose A Player, then Pay \$30000 | P |
| doAction(Player[] players, int turn, int | 9 | Drawn card is a Collect From Player | Drawn card is File a Lawsuit | PLAYER 1's turn! MONEY: 230000.0 | PLAYER 1's turn! MONEY: 230000.0 | P |

| | | | | | | |
|------------------------------------|----|--|--|---|---|---|
| numberOfPlayersInGame) | | card | | LOAN: 0.0 Action card drawn: FILE A LAWSUIT! Bring them to court! Updated current MONEY: \$260000.0 | LOAN: 0.0 Action card drawn: FILE A LAWSUIT! Bring them to court! Updated current MONEY: \$260000.0 | |
| | | Drawn card is a Collect From Bank card | Drawn card is Tax Refund | PLAYER 1's turn! MONEY: 200000.0 LOAN: 0.0 INTEREST: 0.0 Action card drawn: TAX REFUND Get your money back from the taxpayers! \$30000 added to Player 1 Updated MONEY: 230000.0 | PLAYER 1's turn! MONEY: 200000.0 LOAN: 0.0 INTEREST: 0.0 Action card drawn: TAX REFUND Get your money back from the taxpayers! \$30000 added to Player 1 Updated MONEY: 230000.0 | P |
| | | Drawn card is a Pay The Bank card | Drawn card is Traffic Violation | PLAYER 1's turn! MONEY: 200000.0 LOAN: 0.0 INTEREST: 0.0 Action card drawn: TRAFFIC VIOLATION Oopsies! No escaping the law! \$30000 deducted from Player 1 Updated current MONEY: 170000.0 | PLAYER 1's turn! MONEY: 200000.0 LOAN: 0.0 INTEREST: 0.0 Action card drawn: TRAFFIC VIOLATION Oopsies! No escaping the law! \$30000 deducted from Player 1 Updated current MONEY: 170000.0 | P |
| top(ArrayList<ActionCard> deck) | 10 | Return the current top most card set by the head | Head currently points at card WATCH A SHOW | <Method returns card WATCH A SHOW> | <Method returns card WATCH A SHOW> | P |
| pop((ArrayList<ActionCard> > deck) | 11 | Return the current top most card set by the head, head does not point at the last card | Head currently points at card SETUP SCHOOL | <Method returns card SETUP SCHOOL, head is decremented> | <Method returns card SETUP SCHOOL, head is decremented> | P |
| | | Return the current top most card set by the head, head points at the last card | Head currently points at the last card of the Action Card deck | DECK RAN OUT! Cards to be automatically reshuffled at NEXT turn. <Action cards are reshuffled, and head value is resetted> | DECK RAN OUT! Cards to be automatically reshuffled at NEXT turn <Action cards are reshuffled, and head value is resetted>. | P |
| setHead | 12 | Set head at user-specified value within valid range (1-50) | -ac terminal argument 2 | Head is set at 2. | Head is set at 2. | P |
| | | Set head at user-specified value outside valid range | -1 | <Program exits with return status -1> | <Program exits with return status -1> | P |

| | | | | | | |
|---------------|----|----------------------------|---|--|--|---|
| | | (1-50) | | | | |
| getMainID | 13 | Card is TAX REFUND | Current MainID is 0 | Returned value is 0 | Returned value is 0 | P |
| | | Card is HIKING | Current MainID is 1 | Returned value is 1 | Returned value is 1 | P |
| getTypeOfCard | 14 | Card is LAWSUIT | Current typeOfCard value is Choose A Player, then Pay \$30000 | Choose A Player, then Pay \$30000 | Choose A Player, then Pay \$30000 | P |
| | | Card is IT'S YOUR BIRTHDAY | Current typeOfCard value is Collect \$30000 From Each Players | Collect \$30000 From Each Players | Collect \$30000 From Each Players | P |
| toString() | 15 | Card is TRAFFIC VIOLATION | N/A | TRAFFIC VIOLATION Oopsies! No escaping the law! | TRAFFIC VIOLATION Oopsies! No escaping the law! | P |
| | | Card is CHRISTMAS BONUS | N/A | CHRISTMAS BONUS Wish all a happy Merry Christmas! | CHRISTMAS BONUS Wish all a happy Merry Christmas! | P |