Project J: The Donald

A. Introduction

In 2019, there was a group of university students who studied Computer Science who were too bored of studying. One day, they gathered and talked about it. Then, one of them proposed a new game named “The Donald”. It was a fun card game that requires lots of brain cells to be sacrificed if you wanted to win the game. It sounds interesting, and succeeded to attract the attention of the others, but then sadly they found that they didn’t have the important tool to play it which is the card. “Hrmm, we could code this out in java couldn’t we?”, one of them said. “Well we could score fresh blood to play with if we make the juniors do it.”, another kind of bad guy said. “Hehe yes, big brother can teach u how to have fun :)”, they said.

B. Problem Statement

You are required to develop a program that would be able to play the game. The program should be able to be played with multiplayer without the internet.

i) Basic Requirement

The program should support multiplayers. By default, there should be 4 players who would be forming two teams each with 2 people. The program should be able to calculate the score of each player and define the winning point correctly. The game should also follow the rules stated below.

ii) Game rules

There is a card deck with 52 cards, each card with a different Rank or Color. The value of the card is compared by the color then only by the rank value. However, there is one more rule for this, for example if the first player played card BLUE 4, then all the following players should play BLUE card also if they have a BLUE card on their hand. If they don’t have a BLUE card, they can play any other color card, but the other card would be considered as having lowest value unless it is a “donald” card which is called at the beginning **if and only if the player who called for the donald have played the card already**. To have a better understanding, can refer to the output in the output section. Then **after the player has won a round of game, the next round would be starting from the player who won the round**.

Default Card Priority

Rank value : 1 < 2 < 3 < 4 < 5 < 6 < 7 < 8 < 9 < 10 < A < B < C

At the beginning, after the cards are assigned equally and randomly to the players, they should start to call for donald (number of donald + donald value)to decide the winning criteria. Call for Donald would change the winning criteria. For example, if I call for 3 RED, then my team would need to win 9 (6 + 3) pairs of cards to win the game while the other team only needs 5 (8 - 3) pairs of cards to win the game. Besides, if 3 RED is called then it means the RED card would be bigger than any other color card without considering their rank if and only if the RED card is played by the player who called for it. Same goes to other donald values except for NO DONALD which means the game will follow the default card priority. When calling for donald, if someone has called for a certain value like 1 GREEN, then the following player should call for a certain value that is higher than 1 GREEN, for example 2 GREEN , 1 BLUE, 1 YELLOW, or NO DONALD if they want to bid for it. The value of the donald called would be compared by number then only followed by color.

Donald Priority

Donald number: 1 < 2 < 3 < 4 < 5 < 6

Donald color : GREEN = BLUE = YELLOW = RED = NO DONALD

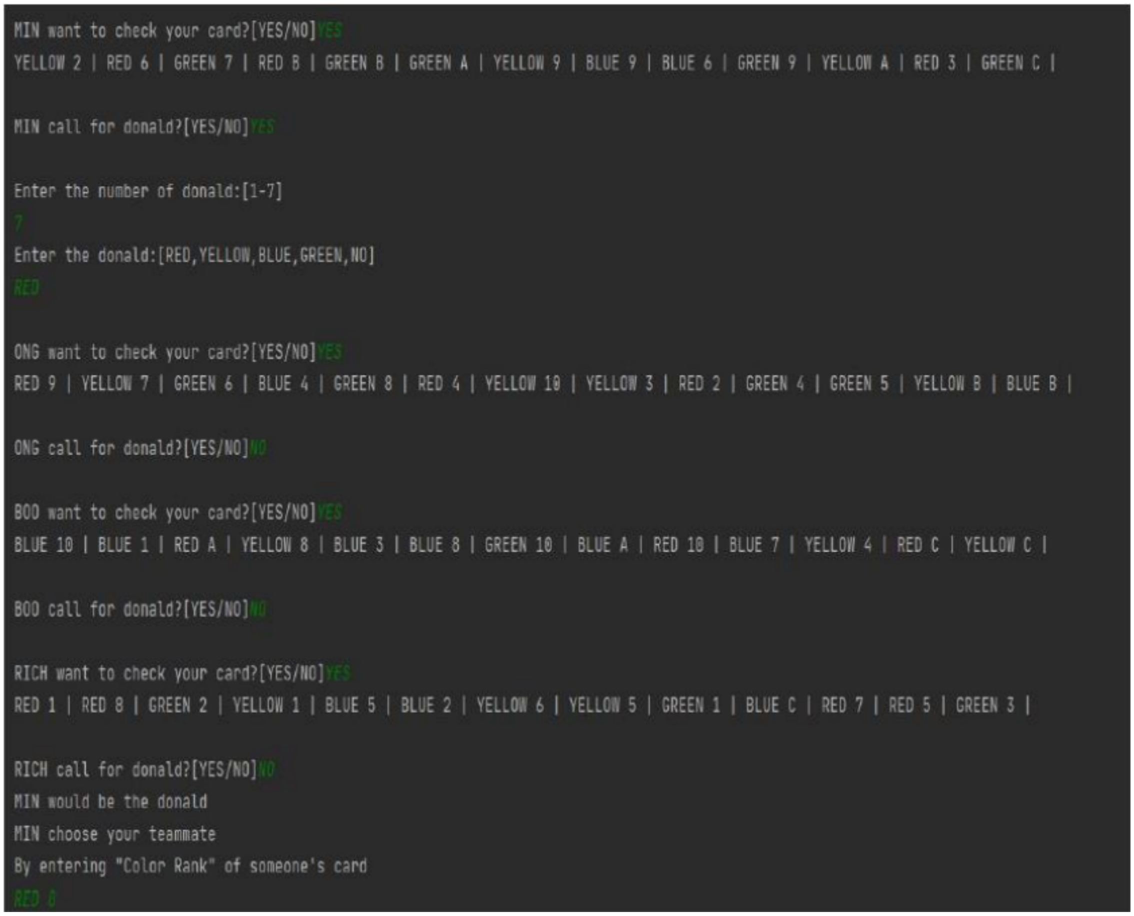
(after the donald color is selected it will become the highest value)

Winning Criteria

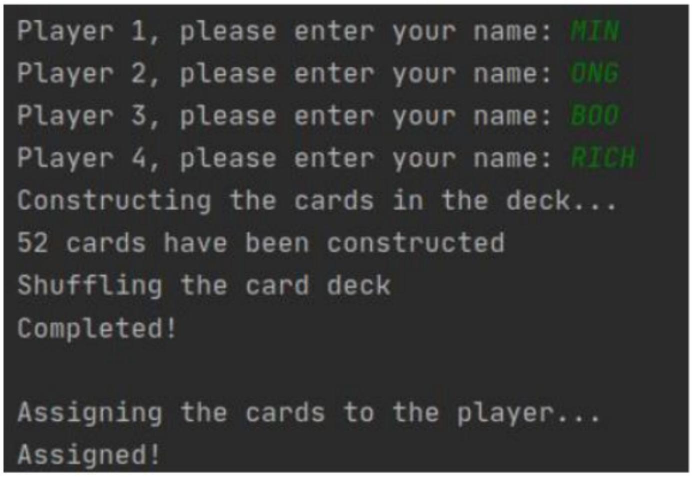
Team who called for donald: 6 + number of donald

Another team: 8 - (number of donald)

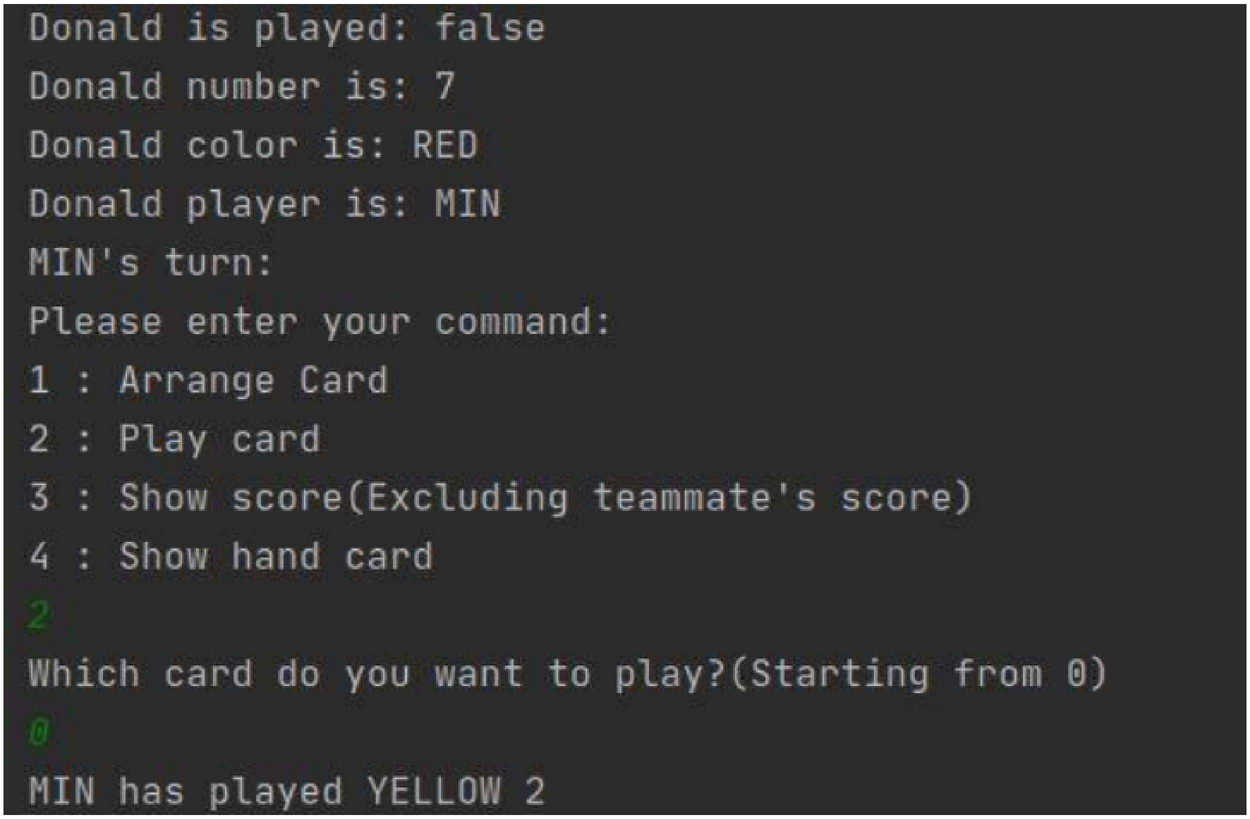
C. Sample



```
MIN want to check your card?[YES/NO]YES
YELLOW 2 | RED 6 | GREEN 7 | RED B | GREEN B | GREEN A | YELLOW 9 | BLUE 9 | BLUE 6 | GREEN 9 | YELLOW A | RED 3 | GREEN C |

MIN call for donald?[YES/NO]YES

Enter the number of donald:[1-7]
7
Enter the donald:[RED,YELLOW,BLUE,GREEN,NO]
RED

ONG want to check your card?[YES/NO]YES
RED 9 | YELLOW 7 | GREEN 6 | BLUE 4 | GREEN 8 | RED 4 | YELLOW 10 | YELLOW 3 | RED 2 | GREEN 4 | GREEN 5 | YELLOW B | BLUE B |

ONG call for donald?[YES/NO]NO

BOO want to check your card?[YES/NO]YES
BLUE 10 | BLUE 1 | RED A | YELLOW 8 | BLUE 3 | BLUE 8 | GREEN 10 | BLUE A | RED 10 | BLUE 7 | YELLOW 4 | RED C | YELLOW C |

BOO call for donald?[YES/NO]NO

RICH want to check your card?[YES/NO]YES
RED 1 | RED 8 | GREEN 2 | YELLOW 1 | BLUE 5 | BLUE 2 | YELLOW 6 | YELLOW 5 | GREEN 1 | BLUE C | RED 7 | RED 5 | GREEN 3 |

RICH call for donald?[YES/NO]NO
MIN would be the donald
MIN choose your teammate
By entering "Color Rank" of someone's card
RED 8
```



```
Player 1, please enter your name: MIN
Player 2, please enter your name: ONG
Player 3, please enter your name: BOO
Player 4, please enter your name: RICH
Constructing the cards in the deck...
52 cards have been constructed
Shuffling the card deck
Completed!

Assigning the cards to the player...
Assigned!
```



```
Donald is played: false
Donald number is: 7
Donald color is: RED
Donald player is: MIN
MIN's turn:
Please enter your command:
1 : Arrange Card
2 : Play card
3 : Show score(Excluding teammate's score)
4 : Show hand card
2
Which card do you want to play?(Starting from 0)
0
MIN has played YELLOW 2
```

Example of the game interface

<u>Situation :</u>

Player 1 called for 7 RED as donald.

Player 1 called for a player who has card RED 8 as the teammate. So in this case, player 1 would team with player 4 and they would require to win (6+7) which is 13 rounds to win the game while player 2 and player 3 would be in a team and they would require to win (8-7) which is 1 round to win the game.

When the first player who would play the card plays let's say BLUE 9, then the other player should play the BLUE card also if they have a BLUE card in their hand.

Scenario 1:

Player 1 plays BLUE 8

Player 2 has a BLUE card, so must play either one of the BLUE cards also. For example, player 2 played BLUE 9

Player 3 has a BLUE card also, so must play either one of them. For example, player 3 played BLUE 10

Player 4 got no BLUE card, then he/she could play any card he/she wants. But if the card played by Player 4 is not a donald card, then the card would be having lowest rank instead. For example, donald cards are RED. player 4 played YELLOW A, it is not a donald card. So yellow A would have the lowest rank in this case.

So in scenario 1, player 3 would win the round. Then, the game ends, player 3 and 2 won the game.

Scenario 2:

Player 1 plays BLUE 8

Player 2 has a BLUE card, so must play either one of the BLUE cards also. For example, player 2 played BLUE 9

Player 3 has a BLUE card also, so must play either one of them. For example, player 3 played BLUE 10

Player 4 got no BLUE card, then he/she could play any card he/she wants. For example, Donald card is RED. player 4 played RED 8, it is donald card. However player 1 (the person who called for donald) never played any RED card before so meaning the donald is not "activated" yet so it is still considered as the lowest rank card.

So in scenario 2, player 3 would win the round. Then, the game ends, player 3 and 2 won the game.
Scenario 3:

Player 1 plays BLUE 8

Player 2 has a BLUE card, so must play either one of the BLUE card also.

Player 3 has a BLUE card also, so must play either one of them

Player 4 got no BLUE card, then he/she could play any card he/she wants. For example, Donald card is RED. player 4 played RED 8, it is donald card and player 1 (the person who called for donald) played RED card before so it is considered as "activated". The RED 8 would be considered as having the highest rank. So in this case, player 4 would win the game.

So in scenario 3, player 4 would win the round. Then, the next round of the game would be started from player 4.

D. Additional Challenge

1) Make some different game modes such as player vs computer.

2) Deploy a GUI (Graphic User Interface) for it.

3) Feel free to add in any other card games into it.

4) Make use of the Exception class to handle the input of users. For example, catch the InputMisMatchException for the field where the user should enter an integer, but he/she gave it a float value.

5) Any other features that you could think of.

E. Submission (Important!)

1) The deadline for this project is 24/5/2021(Monday). Submissions after the deadline will not be evaluated.

2) For the first line of every .java file, write your name using comment line syntax "//" (example: //NAME).

3) Include a ReadMe.txt text file in your submission briefly describing what you have written in your program. If you made your own creative touches to the program, do remember to include them here.

4) After completing the program, compress all your files into a .zip or .rar file. Rename it as NAME.rar or NAME.zip and submit at the submission section in Project J official website or by filling in this google form (link: https://forms.gle/XBHDeL3N5xpQKhNa8).

5} Winners of Project J will be announced during the closing ceremony: 28/5/2021, Friday 8pm.

<u>Some Questions you might have :</u>

1) For bidding the donald number, you can either do conditions for when the players call the same number using conditions for different scenarios or your own implementation.
2) NO DONALD doesn't mean no donald. What I mean is NO DONALD is a kind of DONALD. When NO DONALD Is called, it means there is no color that would have a higher rank than others. For example if it is RED donald, RED card would have a higher rank than other color cards if the donald player played any RED card. So this situation won't happen to NO DONALD. With NO DONALD, if player 1 starts in round 1, if player 1 plays GREEN card, then other player who plays NOT GREEN card will for sure lose the round because it is no DONALD COLOUR so other players must play GREEN card in order to win the round (they must play it also, unless they have no GREEN card in their hand),so the game would just be throwing in the highest card of the same color and if you don't have that color then you've lost the round.

README :

Psst…psssst…hey, Hey you! Yeah you, you need some help with the project thingy A.C.E gave? Yeah you know you do, and I got them good stuffz, here, just follow my lead!

1) Create four classes, DonaldCard, DonaldCardEvaluate, DonaldRoundSimulator, and Scoreboard
2) Create the cards as objects

DonaldCard :

Will include the variables such as card parameters, player names, donald player reference and teammates, references to cards each player will drop each round, the bid number of each player, The list of objects which form each players' hand and a way to present them inside the log, winning criteria for each team, wins of each player, activate status of donald card, list of all cards, and the following methods :

- Randomizer to randomly give players their hands method
- Teammate finder method
- Checks whether a player has a card method
- toString method
- exit method
- constructor
- randomizer for when more than one player bid the same highest number

DonaldCardEvaluate :

- Extends DonaldCard
- Assigning the after-activation value to donald card

DonaldRoundSimulator :

- Extends DonaldCard
- Card finder method (finds the card from the deck based on what player plays)
- Drop card and color checker method for all players
- Inhand method (checks whether a player has a card or not)
- Changes the value of donald when played
- Method to keep the donald value changed for each round after it being activated
- Odd color scenario method
- Comparer of cards and announcing winner of round

Scoreboard :

- Extends DonaldCard
- Finds the number of wins for each team
- Scoreboard method to break the game loop when one team wins