THE MONOPOLY GAME

Task 1 (Exam 1 – first folder):

- Retrieve the title deeds in the file.
- Store the title deeds in an array list (use the structure created).
- Display all the title deeds calling the **displayTitleDeed()** repeatedly.

Task 2 (Exam 2 – second folder and will be used also by Task 3):

- Create a function that will initialize the game board with default values called **initGameBoard()**.
- The function will accept the Game Board instance.

- The owned by bank title deeds will retrieve all titles from the file “titledeed.cis”.
- Sets the board lot into the designated values:
    - Property lots owned by the bank will have zero (0) as value.
    - Index zero is the starting position marked as GO in the board which will be owned by the bank.
    - Community Chest will have the value -1.
    - Chance marked as question mark (?) will have the value -2.
    - Free parking will have the value -3.
    - Income tax will have the value -4.
    - Jail will have the value -5.
    - Electric company will have the value -6.
    - Water works will have the value -7.
    - The transportation will have the value -8, -9, -10, and -11 respectively starting from position 0.
    - Go to jail will have the value -12.
- Set the values for the cash dispenser.
    - Index 0 will be for $500
    - Index 1 will be for $100
    - Index 2 will be for $50
    - Index 3 will be for $20
    - Index 4 will be for $10
    - Index 5 will be for $5
    - Index 6 will be for $1
    - All of the them should have 30 bills each denomination.
- Set the total money based on the cash dispenser.
- The players array will contain the active players of the game initially empty.
- The player count will contain the number of active players.
- The house count is the number of houses available to be leased. This gets subtracted once a player will acquire a house. This gets added when the property owners sell the house.
- The hotel count is the number off hotels available to be leased. When the property owner wants to buy more house after buying 4 houses this will be converted to a hotel. The 4 houses will be retrieve and change into a hotel property.
- Chance and community chest cards are the cards that either brings good or bad luck drawn after the player lands on board lots mark with this. ***(just mark this as empty or no need to implement as it was not yet used for the current version)***
- The top is just the current position of the next card to be drawn. Set as 0.
- The dice is for the two dice values after roll by each of the player that will be used to move the players position in board. Consecutive 3 similar values would land the player automatically to the jail and will skip it’s turn.

Task 3 (Exam 2 – second folder and same as Task 2):

- Create a function that will add a player called **addPlayer()** and insert them at the rear of the players field in the Game Board instance.
- This will accept the Game board instance and the player id.

- The properties owned will be initially be empty.
- The player will initially receive 2pcs-$500, 2pcs-$100, 2pcs-$50, 6pcs-$20, 5pcs-$10, 5pcs-$5, and 5pcs-$1.
- Set the total money.
- The starting position as zero.
- Add 3 players to the current instance of the board by calling your **addPlayer()**. This should adjust necessary values needed.

Task 4 (Exam 3 – third folder):

- Create a function called **acquireProperty()**.
- The function will accept the Game Board, the id of the player, and the id of the title deed.
- Scenarios
    - The title deed is still owned by the bank.
        - The title deed will then be **transferred** into the player's possession of title deeds. Take note that the player's collection of deeds is stored in a **sorted manner** based on the id of the title deed.
        - The operation will only be successful also if the player has the sufficient funds to acquire the title deed.
        - The title acquisition price is based on the rent price.
        - Adjust necessary values in the Game Board including the counters and the money used.
    - The title deed is already owned by the current player but no house or still lesser than 4 houses.
        - Automatically adds one house in the property in possession of the player's properties.
        - The player pays the house cost to the bank.
        - The bank gives the house.
        - Adjust necessary values in the Game Board including the counters and the money used.
        - The operation will not be successful if the player doesn't have sufficient money to acquire for a house.
    - The title deed is already owned by the current player but has 4 houses.
        - Automatically the houses will be converted to a hotel.
        - The player pays the cost to the bank and give back the 4 houses.
        - The bank then gives a hotel.
        - Adjust necessary values in the Game Board including the counters and the money used.
        - The operation will not be successful if the player doesn't have sufficient money to acquire for a hotel.
    - The title deed is already owned by another player.
        - Automatically you can acquire the title deed and its properties times 3 of the current value of the property based on the rent value depending on the lot, number of houses, or hotel present.

        - Adjust necessary values in the Game Board including the counters and the money used.
        - The operation will not be successful if the player doesn't have sufficient money to acquire the title deed and its properties.
- You have the option to create smaller functions for different scenarios presented above.

Task 5 (Exam 4 – fourth folder):

- Create a function called **storeProperty()**.
- The function will store the title deeds owned by a specified player into file.
- The name of file should have the extension cis.
- The name of file should be "player" + the id of the player + "_titledeeds".

Title Deed Information:

The title deed contains the following:

- id
- name of the title deed
- the color of the title deed
- the amounts and cost stored in an array
    - index 0 for rent cost of title
    - index 1 for rent cost for other players with 1 house
    - index 2 for rent cost for other players with 2 houses
    - index 3 for rent cost for other players with 3 houses
    - index 4 for rent cost for other players with 4 houses
    - index 5 for rent cost for other players with a hotel
    - index 6 is the mortgage value to bank
    - index 7 is the acquisition cost for a house
    - index 8 is the acquisition cost for a hotel which is times 5 of the cost for a house in index 7

LIBRARIES

- stdbool.h
    - usage for the values true or false using the datatype **bool**
- string.h
- stdlib.h

- stdio.h