#Homework 2 (14/10)

Rules:

- ✓ Deadline for completion of homework until: 27 October 2021 (23:59, *before midnight*)
- ✓ 10 points will be deducted for one day delay, you will get 0 points if you are 10 days late.
- ✓ You need to do a demo to get the score, Demo date: 21/10, 22/10, 28/10, 29/10
- ✓ Upload your homework on Portal:
  - o Compress your homework folder into .rar / .zip / .7z
  - o Use your student id as the name for your homework, along with the homework code. (Example: 1086412_HW2.rar / 1086412_HW2.zip / 1086412_HW2.7z).

Homework Tasks:

➢ **Task 1 (50%)**
Create a program to shuffle and deal a deck of cards, the program should consist of *class* **Card**, *class* **DeckOfCard**, and *driver program.*
Details:
  - a) *Class* **Card** *(20%)*
    - Inside this class must provide **suit** (*clubs, spade, heart, diamond*) and **face** (*1, 2, 3, 4, 5, 6, 7, 8, 9, 10, Jack, Queen, King*), this two variable must be an **integer** type.
    - A constructor that's receives two integers from suit and face and use them to initialize them.
    - Static arrays of string representing faces and suits.
  - b) *Class* **DeckOfCard** *(20%)*
    - Create a vector with name "deck" to store all of the cards (total 52 cards)
    - Create an integer named "currentCard" representing the next card to deal.
    - A shuffle function that shuffles the cards in the deck. (you can make your own algorithm to shuffle the cards)
  - c) **Driver Program** *(10%)*
    - Should create DeckOfCards object
    - Should call the shuffle function
    - Deal all 52 cards to 1 player (all the cards from deck must be used)

Example Output:
Player:
1_Spade, Jack_Spade, 2_Diamond, 2_Heart,5_King, 9_Spade, *... (until 52 cards)*

➤ Task 2 (20%)

Continue from Task 1, change the program from dealing 52 cards to 1 player to 4 players, don't forget to divide the total of the cards in the deck with total player. Before the program start the user must set how many users that will get the card and set name for each user.

Example Output:

How many Player? (1,2, or 4)

-> 4 --- *(input by user)*

Player 1 Name?

-> Ben --- *(input by user)*

Player 2 Name?

-> John -- *(input by user)*

Player 3 Name?

-> Tina --- *(input by user)*

Player 4 Name?

-> Albert --- *(input by user)*

Total 4 Player

Player 1 (Ben):

1_Spade, Jack_Clubs, 2_Diamond, 2_Heart, 5_Clubs, 9_Spade, *... (until 13 cards)*

Player 2 (John):

8_Heart, 5_Spade, 1_Diamond, 4_Spade,5_Diamond, 2_Clubs, *... (until 13 cards)*

Player 3 (Tina):

King_Spade, Jack_Heart, 3_Diamond, 2_Heart, King_Clubs, 9_Heart, *... (until 13 cards)*

Player 4 (Albert):

1_Clubs, Jack_Spade, 1_Heart, 6_Heart,5_Heart, Queen_Spade, *... (until 13 cards)*

➤ Task 3 (10%)

What is the different between dynamic memory allocation and deallocation operators new, new [], delete and delete []. Compare it with create a simple program.

➤ Task 4 (20%)

Create a simple program with Proxy Classes Implemented, with condition:
- o Create 2 header files (1 header file contain a proxy class, and 1 header connect with it),
- o Create 1 main file that can connect with the headers.