# HACETTEPE UNIVERSITY

# FACULTY OF ENGINEERING - COMPUTER SCIENCE

BBM203 Software Laboratory I

Assignment I

| Student ID | Name & Surname | Due date |
|---|---|---|
| **21827263** | **Murat Çelik** | **19.11.2020** |

| Subject | Programming Language |
|---|---|
| **Arrays** | **C++** |

# Assignment I

## Goal

Learn how to use arrays as data structures in C ++ language

## Subject

Make a game similar to Solitaire. It will be some different rules but same logic.

### What is Solitaire?

It is a single player game where you divide all cards into their own category within a certain system.

### Terminology

Tableau Area: This is the largest part of the game board, which contains 7 slots for piles.

Pile (Tableau Pile): One of the 7 groups of cards that are stacked on each other downwards.

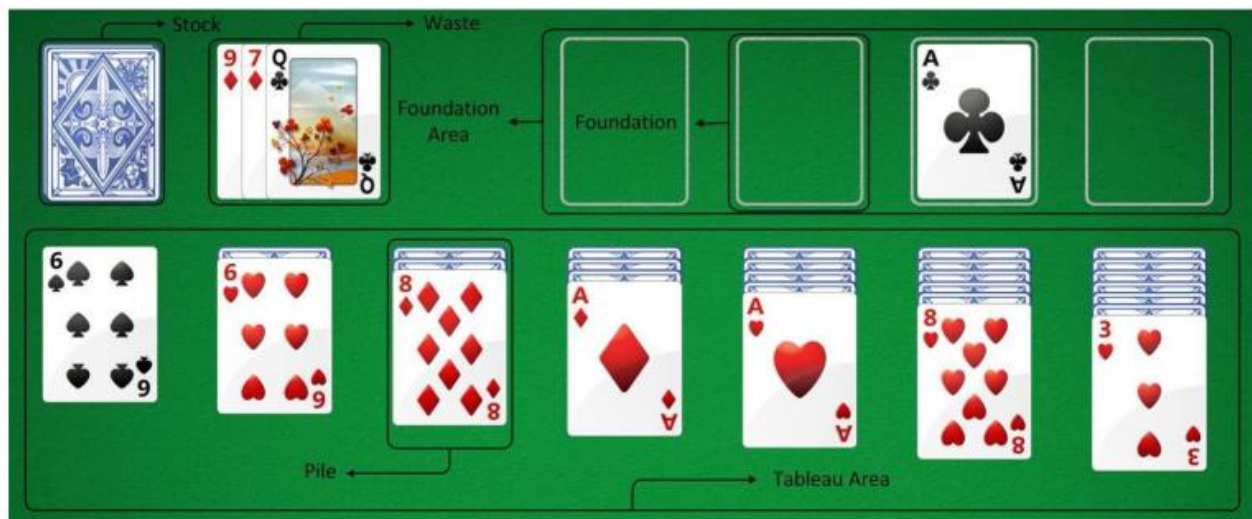Stock: The area where the cards are placed after the cards are dealt.

Waste: The area where we take the card in Stock and put it open

Suit: Card category=> Hearts (♥), Diamonds (♦), Clubs (♣) and Spades (♠).

Foundation Area : The area we put by categorizing all the cards
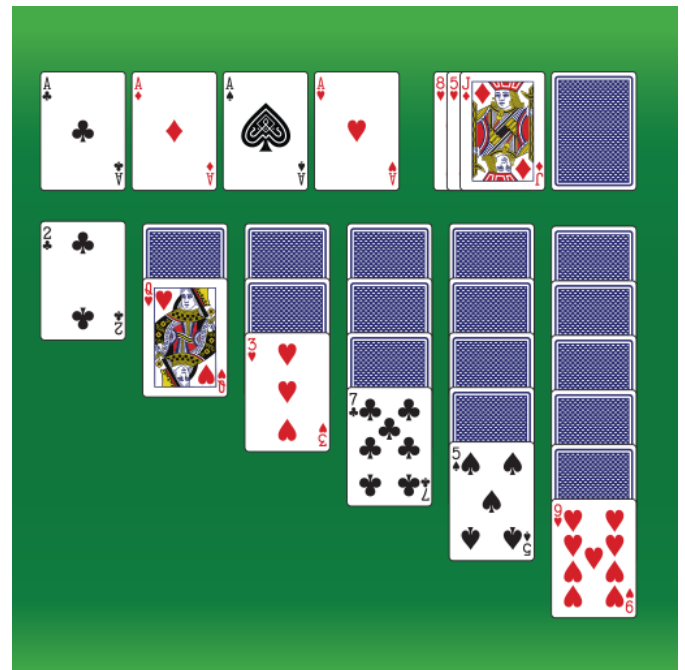
# Solution

## Approach:

First step always be understanding the problem. Therefore, firstly I played Klondike Solitaire. I learnt rules of game. Second step is analyzing problem. When I played the game, I analyzed which one is a object which one is not be object. And I considered to how to be design. After that, I decided what I do, and I was start coding.

## Class Diagram
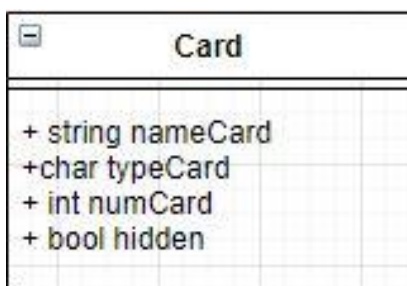


Created object from Card class for each card in deck. Object have four attribute such as: full name, card number, card type, statement of card hidden.

Game class hold arrays of deck. Those arrays have area on the board. Each array have index number. In game time, according to rules of game; some arrays send a Card object and some arrays take the Card object. All operation use these variables.

| Game |
| --- |
| +Card fullDeck[52] |
| +int indexfullDeckindex = 0 |
| +Card pileOne [13] |
| +int pileOneindex = 0 |
| +Card pileTwo [14] |
| +int pileTwoindex = 0 |
| +Card pileThree [15] |
| +int pileThreeindex = 0 |
| +Card pileFour [16] |
| +int pileFourindex = 0 |
| +Card pileFive [17] |
| +int pileFiveindex = 0 |
| +Card pileSix [18] |
| +int pileSixindex = 0 |
| +Card pileSeven [19] |
| +int pileSevenindex = 0;class Card waste [24] |
| +int wasteindex = 0 |
| +int wasteStatement = 0 |
| +class Card foundationOneH[13] |
| +int foundationOneHindex = 0 |
| +class Card foundationTwoD[13] |
| +int foundationTwoDindex = 0 |
| +class Card foundationThreeS[13] |
| +int foundationThreeSindex = 0 |
| +class Card foundationFourC [13] |
| +int foundationFourCindex = 0 |

| Command |
| --- |
| +int movePile(int sourcePile, int numberMove, int goalPile) |
| +int moveWastePile(int numberGoalPile) |
| +int moveFoundationWaste() |
| +int moveFoundationPile(int order) |
| +int openStock() |
| +int openHiddenCard(int number) |
| +void exitCommand() |

Command class have all command operations. These operations was designed according to rules of games. When player moved card, the relevant process is invoked.

Write class have some operations for use write a output. After each command executed, these operations invoked.

| Write |
| --- |
| +void writeFile() |
| +void writeStock() |
| +bool writeWaste() |
| +void writeFoundation() |
| +void writePile() |

## Use Arrays

Arrays are a data structure that keeps elements of the same type in order. In this project, I saved the cards in the array. I have created an array for each field. I used all these arrays in operations with their own index variables.

### How to compile and run?

First : dos2unix * for Dev server

Second: g++ -std=c++11 *.cpp -o main for compile

Third: ./main deck.txt commands.txt output.txt