Assignment 2: Free Cell Game

In this assignment, you are required to implement the Free Cell game using **dynamic arrays**.

You are **not allowed** to use vectors for this assignment.

PLEASE GO THROUGH THE MANUAL THOROUGHLY BEFORE STARTING!

IMPORTANT GUIDELINES:

1) DEADLINE : 25th October
2) Lead TAs: Momina Amer Akhter(22100009) & Armish Javed (24100202)
3) SUBMISSION REQUIREMENTS:
   - Your code file should be zipped in a folder with the following naming convention: PA2_RollNum e.g. PA2_24100123
   - TAs may conduct your viva at any time so that you can explain your code
   - Your assignment will be checked for plagiarism via MOSS.
   - Any case of cheating/plagiarism will strictly be forwaded to DC!
   - Any late assignment or any other file except the above mentioned will not be accepted.
4) CODING CONVENTIONS:

1. Constants are “ALLCAPS” or “ALL_CAPS”.
2. Variables are “allsmall” or “all_small”.
3. All function names must be “firstWordSmallAllOtherWordsCamelCase.
4. All class names must be “CamelCaseWords”.
5. All curly brackets defining a block must be vertically aligned. E.g.

```
for (condition(s))

{

//your code here

}
```

And NOT

```
for (condition(s)){
```

//code}

You are advised to first play the game and understand it before implementing this assignment.

Game Plan:

Free Cell is a single player, one deck card game. In this game, there are:

- 8 columns of cards
- 4 Free cells on the top left corner
- 4 Home slots on the top right corner

When the game starts:

- the first 4 columns have 7 cards each.
- the next 4 columns have 6 cards each.
- free cells and home slots are empty. Each free cell can have one card at a time. Each home slot has to stack up all 13 cards of a single suit starting from Ace to King.

How to play:

- Column to Column Move:
    - You can move a card from one column to another column if the card of source column and card of destination column are of alternating color and source column card is immediate descendent in rank of destination tableau card [except for the Ace card, Ace card should be moved to Home slot]. You can also move multiple cards in one move if the source cards follow the above mentioned rule i.e. after moving multiple cards from source column pile, resulting column will form a descending sequence of cards with alternating colors
- Column to Free Cell Move:
    - You can move any top most/exposed card from any column to any available Free Cell slot except for Ace card. Ace card should be moved to Home slot.
- Free Cell to Column Move:
    - A card from free cell slot can be moved to column card if it follows the condition mentioned in column to column move i.e. card of source Free Cell and card of destination column pile are of alternating color and source Free Cell card is immediate descendent in rank of destination column card.
- Column to Home Slot Move:

    - The first card that will be moved to Home slot will be Ace card. After Ace card, cards will be piled up in Home slot in successive ascending order till King. Each home slot will contain pile of cards of a particular suit only. When all cards will be moved to home slots, player wins the game.
- Free Cell to Home Slot Move:
    - A card from Free Cell can be moved to home slot if the suit of free cell card is same as that of cards (if present) in home slot and free cell card should be in successive ascending order. If any Home slot is empty, the first card that will be moved to Home slot should be Ace card.

Winning Condtion:

The game ends when a player piles up all the cards to home slots i.e. 13 cards of each suit to each home slot.

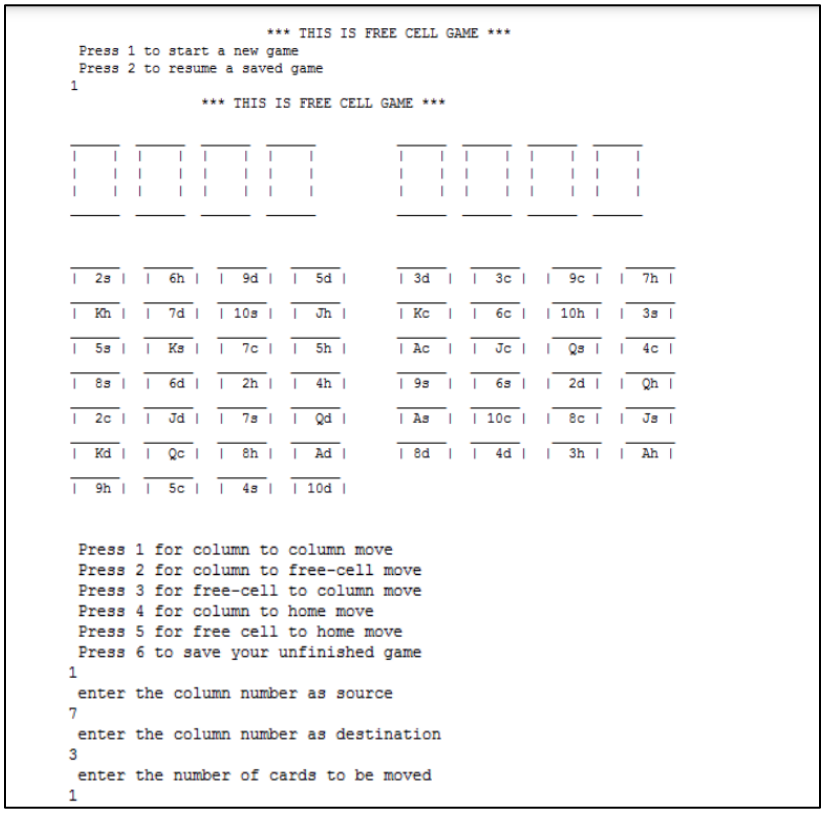
Your Task:

- Use the starter code provided to you and write your code in that file.
- Implement the above functionality described in Section "How to Play." Ensure that with each new game, the cards should be placed randomly in the columns.
- Your game should be able to display the updated state of the game after every move.
- Your game should be able to tell when a player has won the game.
- You have to change the size of your columns dynamically.
- Implement the functionality to save the state of an ongoing game in a file in order to play it later. A player should be able to play a previously saved game.
- Implement error handling while taking inputs from the player.

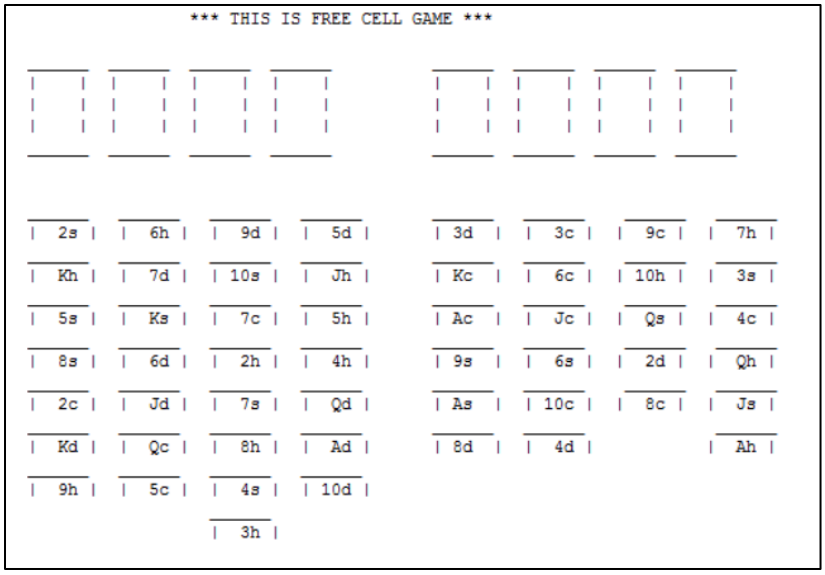
Sample Outputs

Below are some sample outputs to help you understand what we are expecting. However, your interface can differ from this.

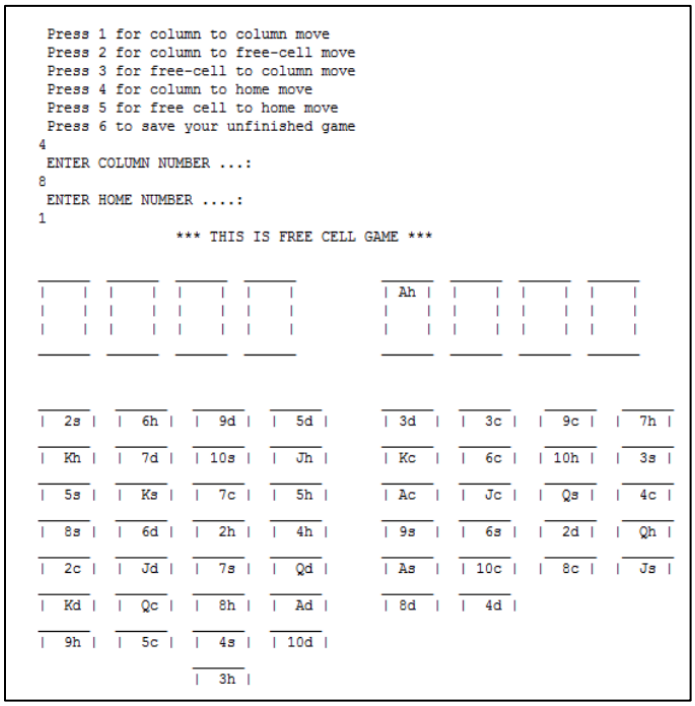
- A column to column move.



```
                    *** THIS IS FREE CELL GAME ***
 Press 1 to start a new game
 Press 2 to resume a saved game
1
              *** THIS IS FREE CELL GAME ***

 _____  _____  _____  _____        _____  _____  _____  _____
|     ||     ||     ||     |      |     ||     ||     ||     |
|     ||     ||     ||     |      |     ||     ||     ||     |
|     ||     ||     ||     |      |     ||     ||     ||     |
 _____  _____  _____  _____        _____  _____  _____  _____


 _____  _____  _____  _____      _____  _____  _____  _____
|  2s |  |  6h |  |  9d |  |  5d |      | 3d  |  |  3c |  |  9c |  |  7h |
 _____  _____  _____  _____      _____  _____  _____  _____
|  Kh |  |  7d |  | 10s |  |  Jh |      | Kc  |  |  6c |  | 10h |  |  3s |
 _____  _____  _____  _____      _____  _____  _____  _____
|  5s |  |  Ks |  |  7c |  |  5h |      | Ac  |  |  Jc |  |  Qs |  |  4c |
 _____  _____  _____  _____      _____  _____  _____  _____
|  8s |  |  6d |  |  2h |  |  4h |      | 9s  |  |  6s |  |  2d |  |  Qh |
 _____  _____  _____  _____      _____  _____  _____  _____
|  2c |  |  Jd |  |  7s |  |  Qd |      | As  |  | 10c |  |  8c |  |  Js |
 _____  _____  _____  _____      _____  _____  _____  _____
|  Kd |  |  Qc |  |  8h |  |  Ad |      | 8d  |  |  4d |  |  3h |  |  Ah |
 _____  _____  _____  _____
|  9h |  |  5c |  |  4s |  | 10d |


 Press 1 for column to column move
 Press 2 for column to free-cell move
 Press 3 for free-cell to column move
 Press 4 for column to home move
 Press 5 for free cell to home move
 Press 6 to save your unfinished game
1
 enter the column number as source
7
 enter the column number as destination
3
 enter the number of cards to be moved
1
```



- A column to home slot move



- A column to free cell move and an invalid move.

```
 Press 1 for column to column move
 Press 2 for column to free-cell move
 Press 3 for free-cell to column move
 Press 4 for column to home move
 Press 5 for free cell to home move
 Press 6 to save your unfinished game
2
 Enter column number as source
7
 Enter number of free-cell
3
                *** THIS IS FREE CELL GAME ***

_____  _____  _____  _____          _____  _____  _____  _____
|    |  |    |  |  8c|  |    |        | Ah |  |    |  |    |  |    |
|    |  |    |  |    |  |    |        |    |  |    |  |    |  |    |
|    |  |    |  |    |  |    |        |    |  |    |  |    |  |    |
_____  _____  _____  _____          _____  _____  _____  _____


_____  _____  _____  _____      _____  _____  _____  _____
|  2s |  |  6h |  |  9d |  |  5d |      | 3d  |  |  3c |  |  9c |  |  7h |
_____  _____  _____  _____      _____  _____  _____  _____
|  Kh |  |  7d |  | 10s |  |  Jh |      | Kc  |  |  6c |  | 10h |  |  3s |
_____  _____  _____  _____      _____  _____  _____  _____
|  5s |  |  Ks |  |  7c |  |  5h |      | Ac  |  |  Jc |  |  Qs |  |  4c |
_____  _____  _____  _____      _____  _____  _____  _____
|  8s |  |  6d |  |  2h |  |  4h |      | 9s  |  |  6s |  |  2d |  |  Qh |
_____  _____  _____  _____      _____  _____           _____
|  2c |  |  Jd |  |  7s |  |  Qd |      | As  |  | 10c |           |  Js |
_____  _____  _____  _____      _____  _____
|  Kd |  |  Qc |  |  8h |  |  Ad |      | 8d  |  |  4d |
_____  _____  _____  _____      _____  _____
|  9h |  |  5c |  |  4s |  | 10d |
                  _____
                  |  3h |

 Press 1 for column to column move
 Press 2 for column to free-cell move
 Press 3 for free-cell to column move
 Press 4 for column to home move
 Press 5 for free cell to home move
 Press 6 to save your unfinished game
5
 Enter Free Cell Number
3
 Enter Home Number
1
 *** INVALID MOVE ***
```

Marks Distribution:

- Random placement of cards. (10 marks)
- Correct implementation for column to column move. (5 marks)
- Correct implementation for column to free cell move. (5 marks)
- Correct implementation for free cell to column move. (5 marks)
- Correct implementation for column to home move. (5 marks)
- Correct implementation for free cell to home move. (5 marks)
- Correctly printing the state of the game after every move. (5 marks)
- Error handling. (5 marks)
- Saving/Loading a game. (10 marks)
- Detecting Winning Condition. (5 marks)
- Proper descriptive variable names + comments. (5 marks)
- Proper indentation. (5 marks)
- Marks will be DEDUCTED if the coding conventions, or submission guidelines are not followed.

Total Marks: 70