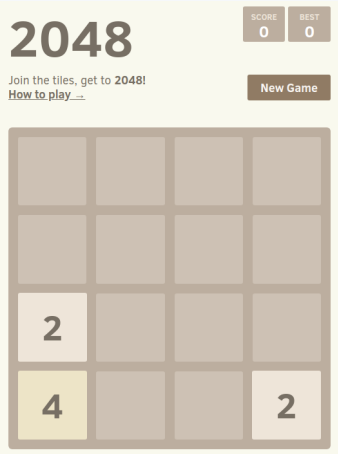
Let's look at a game named '2048'. The game has a 4x4 board that looks something like this:



IThe game starts with one or more random tiles at random positions. The above image is a valid starting point.

According to play2048.co:

Rules of the game

- All the tiles on the board will be a power of 2 like 2, 4, 8, 16, 32, 64, 128,...., 2048.
- There are 4 possible moves: left, right, top, bottom.
- On each move, all the tiles slide in the direction of the move until they are stopped by another tile or an edge.
- A random tile will be inserted at a random empty spot on the board after every move.
- If after sliding, two tiles with the same values collide in the direction of the slide then they will merge into a tile with the value being the total of the collided tiles.
  Example: Two tiles numbered 4 will merge to form a tile numbered 8. The merging will happen in the direction of the movement.
- A merged tile will not merge with another tile in the same move.
- In case 3 consecutive tiles have the same number then the farther tile in the direction of the move will merge. In case all four tiles have the same number

then the first two and last two will merge.

- The game is won if the board has a tile numbered 2048.
- The game is lost if there are no possible moves left: No empty tile and no adjacent tiles with the same number.

Try out the game here for further clarification. Do not try to complete the game as it is highly addictive.

Requirements

Create a command-line application for the above game with the following requirements:

- Initialize the game with two tiles numbered 2 at random positions.
- Print the board after initializing
- Allow the user to make moves.
    - The user will make a move by entering a number.
        - 0 denotes left
        - 1 denotes right
        - 2 denotes top
        - 3 denotes bottom
    - Slide the tiles based on the value, if the slide is possible.
    - Add a random tile on the board
    - Print the board after the move
    - End the game if it is won or lost
        - Print "Congratulations" if the game is won
        - Print "Game over" if the game is lost

Example

Example 1

After initialization, output:

```
- - - -
```

```
- - - -
2 - - -
- - 2 -
```

Input:

```
1
```

Output:

```
- - - -
- - 2 -
- - - 2
- - - 2
```

Input:

```
3
```

Output:

```
- - - -
- - - -
2 - - -
- - 2 4
```

Input:

```
3
```

Output:

```
- - - -
- - - -
- - 2 -
2 - 2 4
```

Input:

```
1
```

Output:

```
- - - -
- - - -
- 2 - 2
- - 4 4
```

Input:

```
1
```

Output:

```
- - - -
- - - -
- - - 4
- - 2 8
```

.

.

.

.

.

After some point, output:

```
- 2 - -
- - 4 2
2 4 4 8
256 512 512 1024
```

Input:

```
1
```

Output:

```
- 2 - -
- - 4 2
2 4 4 8
2 256 1024 1024
```

Input:

```
1
```

Output:

```
- - - 2
2 - 4 2
- 2 8 8
- 2 256 2048
```

```
Congratulations
```

Example 2

After some point, output:

```
2 4 8 -
64 128 32 16
4 8 2 4
1024 512 256 128
```

Input:

```
1
```

Output:

```
4 2 4 8
64 128 32 16
4 8 2 4
1024 512 256 128
Game over
```

Expectations

- **Make sure that you have a working and demonstrable code**
- **Make sure that the code is functionally correct**
- **Code should be modular and readable**
- **Separation of concern should be addressed**
- **Please do not write everything in a single file (if not coding in C/C++)**
- **Code should easily accommodate new requirements and minimal changes**
- **There should be a main method from where the code could be easily testable**
- **[Optional] Write unit tests, if possible**

- No need to create a GUI

Optional Requirements

Please do these only if you've time left. You can write your code such that these could be accommodated without changing your code much.

- Allow the user to play beyond 2048.
- Take the grid size from the user during initialization. The grid could be 4*4 or anything else.
- Allow the user to set the winning value during initialization.
- Allow the user to set the values to be the power of an arbitrary base value during initialization. In the generic version, the base value is 2.
- Allow the user to restart the game after the game ends.