Project Plan for Master Mind :: Escape the Room

Jiaying Wu

How to play Master Mind :: Escape the Room

The four text paragraph display below will separate into two parts:

1. Title start with "~" and end with "~", display when calling the function **`displayTitle()`**.
2. The rest is the body, it will store in different text file.

```
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
                      MasterMind :: Escape the Room
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

"You have been locked in an empty dark room, running out of food and water.
In front of you, there is a gate with a giant lock. The times you can enter
the numbers into the lock is limited! Once you ran out of the times, the
gate will lock forever! Hurry up! Clock is ticking......"

If you enter all correct elements with correct position into the lock in the
limited times, the gate will open. That you can escape the room. Otherwise,
you will be lock in the room forever......

 [N] New game.

 [C] Continue the saved game.

 [H] for Help.

 [A] view the Achievements.

 [E] End the game.

 Enter your game option:
```

The above text is the main page of the game, it will stored in this text file **`masterMindMainPage.txt`**. It will load and display at the beginning of the game for player to read.

```
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
                      MasterMind :: Escape the Room
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

Three levels of this game:

 [1] wood gate = easy, 4 elements out of 6 with 10 times to try.

 [2] Rock gate = tricky: 5 elements out of 8 with 12 times to try.

 [3] Iron gate = hard: 6 elements out of 10 with 14 times to try.

After you enter the elements into the lock, the lock will provide you some clues.

- One `#` will show up if you enter a correct element in correct position.
```

```
- One `*` will show up if you enter a correct element in wrong position.

- No symbol for a worng element

- The order of symbols to show up is random in each time.

The way to caculate the points for each game:

1. The points for wood gate start from 100, rock gate start from 180 and
   iron gate start from 280.

3. Each time will cost you 10 points in wood gate, 15 points in rock gate,
   20 points in iron gate.

4. 2 points for one `#`, 1 point for one `*`.

5. Maximum points for opening wood gate is 98, that is four `#` in 1 time.
   Maximum 175 points for rock gate and maximum 272 points for iron gate.

6. The points you get from each game will add up to your total points.

Ranking tier: 3 level of ranking, player promotion/demotion every 5 games won/lost.

1. Beginner

2. Expert

3. Master

                    Press any key to continue...
```

The above text is the game rule, it will stored in this text file **masterMindHelp.txt**. It will load and display when user press [H] for help.

```
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
                      MasterMind :: Escape the Room
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

Three steps to start the game:

1. Enter your name.

2. Select the gate you want to open:

 [1] Wood Gate = easy: 4 elements out of 6 with 10 times to try.

 [2] Rock Gate = tricky: 5 elements out of 8 with 12 times to try.

 [3] Iron Gate = hard: 6 elements out of 10 with 14 times to try.

3. Select the type of element for the lock:

 [1] Number
```

```
 [2] Letter

 [3] Symbol

 [4] Word

Then you will face to the gate you have chosen.

                        Press any key to continue...
```

The above text is the setup for the game, it will stored in this text file **masterMindStart.txt**. It will load and display when user press [S] to start a new game.

```
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
                        MasterMind :: Escape the Room
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
--------------------------------------------------------------------------------
Player name:
Rank:
Gate attempted:
Gate opened:
Total points:
Highest points in Wood Gate: 0 / 98
Highest points in Rock Gate: 0 / 175
Highest points in Iron Gate: 0 / 272
--------------------------------------------------------------------------------

                         Press any key to continue...
```

This is the initial version of player's achievement. It will stored all players' achievement, identify by player name. It will store in this text file **masterMindAchievement.txt**, it will load and display when player press [A]

Development outline of Master Mind :: Escape the Room

Outline the functionality of all game classes

After reading the assignment brief, I decided to use 3 classes for this game: a Player class, a Code class, and the Application file.

1. Player class

Player class stores the information about the player: their name, the difficulty level and element type that player have chosen, and the game point of player. With difficulty level and element type, it can determine other variable of code.

2. Code class

Code class to generate the secret code, and create variable base on difficulty level and element type. To construct the array to record the code that player input.

3. Application file

Application file will be where all the functionality of the game is stored. Using best practice techniques will ensure that I have well-designed game that is easy for me to debug and test, and is easily read and understood by other programmers. How this structured is covered in the following breakdown of the program.

Six sections for game setup

Section 1: Variables and functions for setup

Variables:

1. *string* `previousOption` to stored player's previous option.
2. *string* `currentOption` to stored player's current option.
3. *string* `playerName` to store player's name.
4. *int* `difficultyLevel` to store the chosen difficulty level of the game.

Functions:

1. *void* `displayTitle()` to display the title of the game.
2. *void* `readTextFile`(*string* `fileName`) to read the given text file and display into screen.
3. *string* `askForString`(*string* `question`) to ask user enter the option, return a *string* value. *string* `question` is a local variable in function `askForString()`.
4. *int* `askForInt`(*string* `question`) to ask user enter the option, return a *int* value. *string* `question` is a local variable in function `askForInt()`.

Section 2: Main page

The main page display every time we start the game, or press [`M`] during the game.

1. Clear the screen.
2. Call the function `displayTitle()` to display the title.
3. Load and display the text file `masterMindMainPage.txt` using function`readTextFile()`.
4. Replace `previousOption` = "m".
5. Call the function `askForString()` to ask player enter their option, and check the input. If not in {"s", "c", "h", "a", "e"}, return error message. Call the function again until player enter the correct option.
6. Store the string return from function `askForString()` as `currentOption`.
7. Then go the page `currentOption` match. Eg, "h" for help page.

Since the screen will pause until the player enter the option, it provide time for player to read the information.

What we need:

Variables:

1. *string* `previousOption`
2. *string* `currentOption`

Functions:

1. *void* `displayTitle()` to display the title.
2. *void* `readTextFile`(*string* `fileName`) to read `masterMindMainPage.txt` then display main page.
3. *string* `askForString`(*string* `question`) to ask a string from player.

Text file:

1. `masterMindMainPage.txt`

Section 3: Help page

The help page display when player press [H].

1. Clear the screen.
2. Call the function **displayTitle()** to display the title.
3. Load and display the text file **masterMindHelp.txt** using function **readTextFile()**.
4. Replace the **previousOption** = **currentOption**.
5. Replace **currentOption** = "h".
6. Include a "Press any key to continue", pause the screen to allow player to read the information.
7. Then go back to the page **previousOption** match. Eg, "m" for main page...

What we need:

Variables:

1. *string* **currentOption**
2. *string* **previousOption**

Functions:

1. *void* **displayTitle()** to display the title.
2. *void* **readTextFile**(*string* **fileName**) to read **masterMindHelp.txt** then display Help page.

Text file:

1. **masterMindHelp.txt**

Section 4: Achievement page

The achievement page display when player press [A].

1. Clear the screen.
2. Call the function **displayTitle()** to display the title.
3. Load and display the text file **masterMindAchievement.txt** using function **readTextFile()**.
4. Replace the **previousOption** = **currentOption**.
5. Replace **currentOption** = "a".
6. Include a "Press any key to continue", pause the screen to allow player to read the information.
7. Then go back to the page **previousOption** match. Eg, "m" for main page...

What we need:

Variables:

1. *string* **currentOption**
2. *string* **previousOption**

Functions:

1. *void* **displayTitle()** to display the title.
2. *void* **readTextFile**(*string* **fileName**) to read **masterMindAchievement.txt** then display achievement page.

Text file:

1. `masterMindAchievement.txt`

Section 5: Start game page

Before the game start, player need to finish some setup.

```
Enter your name: Jiaying

Gate level: [1] Wood Gate = easy  [2] Rock Gate = tricky  [3] Iron Gate = hard

Select a level of the gate: 1

Lock type: [1] Number  [2] Letter  [3] Symbol  [4] Word

Select a type of the lock: 1

                        Press any key to continue...
```

How to do that?

1. Clear the screen.
2. Replace the `previousOption` = `currentOption`.
3. Replace `currentOption` = "n".
4. Call the function `displayTitle()` to display the title.
5. Call the function `askForString()` to ask player's name, store the return *string* as `playerName`.
6. Call the function `askForInt()` to ask player to select the gate to open, that is the game difficulty level.
7. check the input, if it within {1, 2, 3}. Store the return *int* as `difficultyLevel`. Otherwise return error message, then call the function `askForInt()` until player enter the correct integer.
8. Call the function `askForInt()` to ask player select the type of element.
9. check the input, if it within {1, 2, 3, 4}. Store the return *int* as `elementType`. Otherwise return error message, then call the function `askForInt()` until player enter the correct integer.
10. After player finish enter their name and game option, include a "Press any key to continue" in the end.

What we need:

Variables:

1. *string* `currentOption`
2. *string* `previousOption`
3. *string* `playerName`
4. *int* `difficultyLevel`
5. *int* `elementType`

Functions:

1. *void* `displayTitle()` to display the title of the game.
2. *string* `askForString`(*string* `question`) to ask player's name, *string* **question** is the local variable.

3. *int* `askForInt`(*string* `question`) to display the question and ask user's choice, *string* **question** is the local variable.

Section 6: Game page

After player press [N] and enter their name, select game difficulty level and element type, press a key to continue at last. Then shift to the page display below.

```
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
                        MasterMind :: Escape the Room
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

Welcome, Jiaying

 [H] Help    [M] Main page    [N] New game    [E] End game

 .~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~.
 | .---. .---. .---. .---.               |
 | |   | |   | |   | |   |  Hidden Code  |
 | '---' '---' '---' '---'               |
 '~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~'
```

How to do that?

1. Clear the screen.
2. Call the function `displayTitle()` to display the title.
3. Create and initialize an *int* `gameRound` = 0. That indicate the current game round.
4. Create a *bool* `isGameOver` = false, to indicate whether the game is over or not.
5. Create an *int* `codeColumn` = 3 + `difficultyLevel` to indicate the number of columns in one row, that is the numbers of element in the secret code. Possible value is {4, 5, 6}.
6. Create an *int* `numberPossibleElement` = 4 + 2 x `difficultyLevel` to indicate the number of all possible elements in one row. Possible value is {6, 8, 10}.
7. Create an *int* `codeRow` = 8 + 2 * `difficultyLevel` to indicate the number of rows, that is the times players can try. Possible value is {10, 12, 14}.
8. Create and initiate a 2 dimensions *string* array **inputCode**[`codeRow`][`codeColumn`] to the code input by player. Numbers of rows determine by `codeRow`, numbers of columns determine by `codeColumn`.
9. Generate the *string* array `secretCode[codeColumn]` using function `generateSecretCode()`.
10. Create a function *string* `gameTableTitle`() to generate the row of welcome, options and hidden code. *string* `playerName` determine player name, *int* `codeColumn` determine how many cell and `elementType` to determine how many empty space inside the cell.
11. Store the return *string* as `gameTable`, then display `gameTable`.

What we need:

Variables:

1. *string* `playerName`
2. *int* `difficultyLevel`
3. *int* `elementType`

4. *int* `gameRound`
5. *int* `codeColumn`
6. *int* `numberPossibleElement`
7. *int* `codeRow`
8. *string* `inputCode[codeRow][codeColumn]`
9. *string* `secretCode[codeColumn]`
10. *string* `gameTable`
11. *bool* `isGameOver`

Functions:

1. *void* `displayTitle()` to display the title of the game.
2. *string* `generateSecretCode`(*int* `codeColumn`, *int* `elementType`)
3. *string* `gameTableTitle`(*string* `playerName`, *int* `codeColumn`, *int* `elementType`)

The player's turn

```
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
                        MasterMind :: Escape the Room
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

Welcome, Jiaying

 [H] Help    [M] Main page    [N] New game    [E] End game

 .~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~.
 | .---. .---. .---. .---.               |
 | |   | |   | |   | |   |  Hidden Code  |
 | '---' '---' '---' '---'               |
 '~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~'

 The code elements: 0 1 2 3 4 5

 Enter code or the game option: 1111
```

How to do that?

1. `gameRound` += 1
2. Create a function *string* `displayElement`(*int* `elementType`, *int* `numberPossibleElement`) to display all possible elements of number or symbol or letter in of the code base on the difficulty level. Also, read the `masterMindWord.txt` file and display all possible element of word.
3. Store the return *string* as `possibleElement`.
4. Create a function *string* `askForCode`(*string* `question`, *string* `possibleElement`) to ask player enter the code or game option.
5. Check the player input, if not match `possibleElement` or game option, return error message and call `askForCode` again until player enter correct input.

6. Store the return *string* into the *string* array `inputCode[codeRow][codeColumn]`. That is the first row of the array.

What we need:

Variables:

1. *int* `gameRound`
2. *int* `elementType`
3. *int* `numberPossibleElement`
4. *string* `possibleElement`
5. *string* `inputCode[codeRow][codeColumn]`.

Functions:

1. *string* `displayElement`(*int* `elementType`)
2. *string* `askForCode`(*string* `question`, *string* `possibleElement`)

Text file:

1. `masterMindWord.txt`

Processing player input

For example, player have enterd a code "1111", the secret code is "4321". That is one number correctly matched. Hence, return a "`#`" on the back.

```
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
                        MasterMind :: Escape the Room
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

Welcome, Jiaying

 [H] Help    [M] Main page    [N] New game    [E] End game

 .~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~.
 | .---. .---. .---. .---.               |
 | |   | |   | |   | |   |  Hidden Code  |
 | '---' '---' '---' '---'               |
 '~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~'
 | .---. .---. .---. .---.               |
 | | 1 | | 1 | | 1 | | 1 |  #            |
 | '---' '---' '---' '---'               |
 '~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~'

 The code elements: 0 1 2 3 4 5

 Enter the code or game option:
```

How to da that?

1. Clear the screen.
2. Call the function `displayTitle()` to display the title.

3. Create a function *string* `gameTableBody`(*string* `secretCode[codeColumn]`, *string* `inputCode[codeRow][codeColumn]`) to check the each row of input code in `inputCode[codeRow][codeColumn]` against the `secretCode[codeColumn]`. Provide the body part of the table, that is:

```
| .---. .---. .---. .---.               |
| | 1 | | 1 | | 1 | | 1 |  #           |
| '---' '---' '---' '---'               |
'~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~'
```

4. Then `gameTable` += the *string* return by function `gameTableBody()`.
5. Display `gameTable`.
6. Also check whether the input match the secret code, if so *bool* `isGameOver` = true. Then check *int* `gameRound` equal to *int* `codeColumn`, if so *bool* `isGameOver` = true.

What we need:

Variables:

1. `gameTable`
2. *bool* `isGameOver`
3. *int* `gameRound`
4. *int* `codeColumn`

Functions:

1. *void* `displayTitle()`
2. *string* `gameTableBody`(*string* `secretCode[codeColumn]`, *string* `inputCode[codeRow][codeColumn]`)

Providing feedback to player

```
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
                        MasterMind :: Escape the Room
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

Welcome, Jiaying

 [H] Help    [M] Main page    [N] New game    [E] End game

 .~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~.
 | .---. .---. .---. .---.               |
 | |   | |   | |   | |   |  Hidden Code  |
 | '---' '---' '---' '---'               |
 '~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~'
 | .---. .---. .---. .---.               |
 | | 1 | | 1 | | 1 | | 1 |  #           |
 | '---' '---' '---' '---'               |
 '~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~'
 | .---. .---. .---. .---.               |
 | | 4 | | 3 | | 2 | | 1 |  # # # #     |
 | '---' '---' '---' '---'               |
 '~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~'

 Congratulations, Jiaying. Now you have opened the gate and escape the room!
```

```
 You point of this game is: 90

--------------------------------------------------------------------------------
Player name: Jiaying
Rank: Beginner
Gate attempted: 1
Gate opened: 1
Total points: 90
Highest points in Wood Gate: 90 / 98
Highest points in Rock Gate: 0 / 175
Highest points in Iron Gate: 0 / 272
--------------------------------------------------------------------------------

 Enter the game option:
```

How to do that?

1. If `isGameOver` = true.
2. display proper text regarding to the result.
3. Create a function *int* `generatePoint`(*int* `gameRound`, *string* `secretCode[codeColumn]`, *string* `inputCode[codeRow][codeColumn]`) and store it as *int* `currentGamePoint`.
4. Create a function *string* `generateAchievement`(*string* `playerName`, *int* `difficultyLevel`, *int* `currentGamePoint`) to update ateachievement and display it.

What we need:

Variables:

1. *bool* `isGameOver`
2. *int* `gameRound`
3. *string* `secretCode[codeColumn]`
4. *string* `inputCode[codeRow][codeColumn]`
5. *string* `playerName`
6. *int* `difficultyLevel`
7. *int* `currentGamePoint`

Functions:

1. *int* `generatePoint`(*int* `gameRound`, *string* `secretCode[codeColumn]`, *string* `inputCode[codeRow][codeColumn]`)
2. *string* `generateAchievement`(*string* `playerName`, *int* `difficultyLevel`, *int* `currentGamePoint`)

The end game conditions

Player can enter [E] to end the game at anytime.

1. In the middle of the game

```
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
                      MasterMind :: Escape the Room
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

Welcome, Jiaying
```

```
 [H] Help    [M] Main page    [N] New game    [E] End game

 .~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~.
 | .---. .---. .---. .---.               |
 | |   | |   | |   | |   |  Hidden Code  |
 | '---' '---' '---' '---'               |
 '~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~'
 | .---. .---. .---. .---.               |
 | | 1 | | 1 | | 1 | | 1 |  #            |
 | '---' '---' '---' '---'               |
 '~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~'

 The code elements: 0 1 2 3 4 5

 Enter the code or game option: e

 You haven't finish the game, do you want to saved it for next time? (y/n): y

 You have saved this game.

 Thank you for playing this game. Goodbye, Jiaying!
```

1. Call the function *string* **askForString**(*string* **question**) to ask player wether they want to store the uncompeted game.
2. create a function **SaveData()** to store the variable *string* **playerName**, *int* **difficultyLevel**, *int* **elementType**, *string* **secretCode[codeColumn]** and *string* **inputCode[codeRow][codeColumn]**.
3. Display the proper text for ending the game.

What we need

Function:

1. *string* **askForString**(*string* **question**)
2. **SaveData**(*string* **playerName**, *int* **difficultyLevel**, *int* **elementType**, *string* **secretCode[codeColumn]**, *string* **inputCode[codeRow][codeColumn]**)

2. After player finish the game

```
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
                        MasterMind :: Escape the Room
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

Welcome, Jiaying

 [H] Help    [M] Main page    [N] New game    [E] End game

 .~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~.
 | .---. .---. .---. .---.               |
 | |   | |   | |   | |   |  Hidden Code  |
 | '---' '---' '---' '---'               |
 '~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~'
 | .---. .---. .---. .---.               |
 | | 1 | | 1 | | 1 | | 1 |  #            |
 | '---' '---' '---' '---'               |
```

```
 '~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~'
 | .---. .---. .---. .---.             |
 | | 4 | | 3 | | 2 | | 1 |  # # # #    |
 | '---' '---' '---' '---'             |
 '~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~'

 Congratulations, Jiaying. Now you have opened the gate and escape the room!

--------------------------------------------------------------------------------
Player name: Jiaying
Rank: Beginner
Gate attempted: 1
Gate opened: 1
Total points: 90
Highest points in Wood Gate: 90 / 98
Highest points in Rock Gate: 0 / 175
Highest points in Iron Gate: 0 / 272
--------------------------------------------------------------------------------

 Enter the game option: e

 Thank you for playing this game. Goodbye, Jiaying!
```

How to do that?

1. Replace the `previousOption` = `currentOption`.
2. Call function `askForString()` to ask player to enter game option, check again input.
3. Store the return *string* as `currentOption` = "e".
4. Then end the game.

What we need:

Variables:

1. *string* `previousOption`
2. *string* `currentOption`

Function:

1. *string* `askForString`(*string* `question`) to ask player's name, *string* **question** is the local variable.

Additional features included

1. This game have a theme, that is the player need to figure out the code of the lock to escape the room.

2. Player can restored the uncompleted game, that is when player press [M], [N] and [E] if they didn't finish the game.

```
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
                      MasterMind :: Escape the Room
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

Welcome, Jiaying
```

```
 [H] Help    [M] Main page    [N] New game    [E] End game
```

Then, player will be ask the question below, player decide whether to save the game base on their choice.

```
 You haven't finish the game, do you want to saved it for next time? (y/n): y

 You have saved this game.

 Thank you for playing this game. Goodbye, Jiaying!
```

3. Player can continue the saved uncompleted game, that is when they press [C] in the Main page. Then, player will the saved game.

```
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
                      MasterMind :: Escape the Room
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

Welcome, Jiaying

 [H] Help    [M] Main page    [N] New game    [E] End game

 .~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~.
 | .---. .---. .---. .---.               |
 | |   | |   | |   | |   |  Hidden Code  |
 | '---' '---' '---' '---'               |
 '~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~'
 | .---. .---. .---. .---.               |
 | | 1 | | 1 | | 1 | | 1 |  #            |
 | '---' '---' '---' '---'               |
 '~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~'

 The code elements: 0 1 2 3 4 5

 Enter the code or game option:
```

4. Player can press [H] to ask for help, that is to display the game rule.

```
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
                      MasterMind :: Escape the Room
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

Three levels of this game:

 [1] wood gate = easy, 4 elements out of 6 with 10 times to try.

 [2] Rock gate = tricky: 5 elements out of 8 with 12 times to try.

 [3] Iron gate = hard: 6 elements out of 10 with 14 times to try.

After you enter the elements into the lock, the lock will provide you some clues.

- One `#` will show up if you enter a correct element in correct position.

- One `*` will show up if you enter a correct element in wrong position.

- No symbol for a worng element
```

```
- The order of symbols to show up is random in each time.

The way to caculate the points for each game:

1. The points for wood gate start from 100, rock gate start from 180 and
   iron gate start from 280.

3. Each time will cost you 10 points in wood gate, 15 points in rock gate,
   20 points in iron gate.

4. 2 points for one `#`, 1 point for one `*`.

5. Maximum points for opening wood gate is 98, that is four `#` in 1 time.
   Maximum 175 points for rock gate and maximum 272 points for iron gate.

6. The points you get from each game will add up to your total points.

                     Press any key to continue...
```

After player press a key, it will go back to the ongoing game.

```
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
                       MasterMind :: Escape the Room
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

Welcome, Jiaying

 [H] Help    [M] Main page    [N] New game    [E] End game

 .~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~.
 | .---. .---. .---. .---.               |
 | |   | |   | |   | |   |  Hidden Code  |
 | '---' '---' '---' '---'               |
 '~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~'
 | .---. .---. .---. .---.               |
 | | 1 | | 1 | | 1 | | 1 |  #            |
 | '---' '---' '---' '---'               |
 '~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~'

 The code elements: 0 1 2 3 4 5

 Enter the code or game option:
```

5. Press [A] to view the achievements of all players, the identifier is `playerName`. It record the total points of each player and the highest points each player have got in each gate.

```
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
                       MasterMind :: Escape the Room
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
--------------------------------------------------------------------------------
Player name: Jiaying
Rank: Beginner
Gate attempted: 1
Gate opened: 1
Total points: 90
```

```
Highest points in Wood Gate: 90 / 98
Highest points in Rock Gate: 0 / 175
Highest points in Iron Gate: 0 / 272
--------------------------------------------------------------------------------

                        Press any key to continue...
```

6. Player can select difficulty level, that is related to different gate. Including:

- Wood Gate = easy
- Rock Gate = tricky
- Iron Gate = hard

7. There is a score system to estimate points for each game. The rule how to calculate the points have also display in the Help page.

8. Player can select different type of elementto use, including:

- Number
- Symbol
- letter
- Word

9. Since it include the word type of the element, it also need to read word list form a file and store it.

10. Display the board using ASCII art.

11. Player promotion/demotion every 5 games won/lost.

UML class diagrams

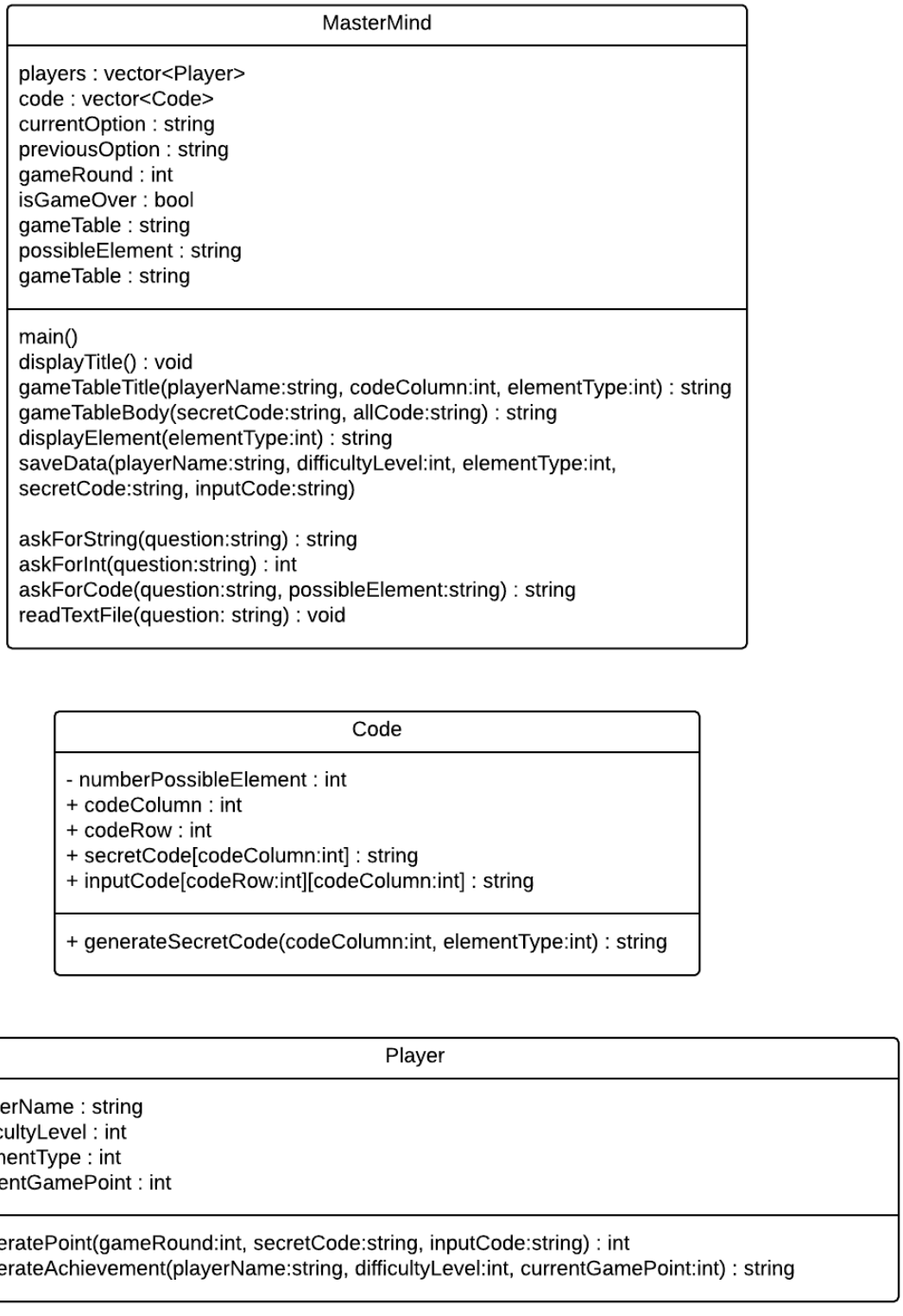
MasterMind
players : vector<Player>
code : vector<Code>
currentOption : string
previousOption : string
gameRound : int
isGameOver : bool
gameTable : string
possibleElement : string
gameTable : string
main()
displayTitle() : void
gameTableTitle(playerName:string, codeColumn:int, elementType:int) : string
gameTableBody(secretCode:string, allCode:string) : string
displayElement(elementType:int) : string
saveData(playerName:string, difficultyLevel:int, elementType:int,
secretCode:string, inputCode:string)
askForString(question:string) : string
askForInt(question:string) : int
askForCode(question:string, possibleElement:string) : string
readTextFile(question: string) : void
Code
- numberPossibleElement : int
+ codeColumn : int
+ codeRow : int
+ secretCode[codeColumn:int] : string
+ inputCode[codeRow:int][codeColumn:int] : string
+ generateSecretCode(codeColumn:int, elementType:int) : string
Player
+ playerName : string
+ difficultyLevel : int
+ elementType : int
+ currentGamePoint : int
+ generatePoint(gameRound:int, secretCode:string, inputCode:string) : int
+ generateAchievement(playerName:string, difficultyLevel:int, currentGamePoint:int) : string

Figure 1: