

# CS101- Algorithms and Programming I

## Lab 10

---

**Lab Objectives:** Arrays. Classes and objects.

---

For all labs in CS 101, your solutions must conform to the CS101 style guidelines (rules!)

### Part 1:

1. Create an application, `Lab10_Q1.java` that does the following:
  - a. Implement a method, `makeAdjacent()`, that takes an array of `int` as a parameter, and returns an array that contains the same numbers as the given array but rearranged so that every 4 is immediately followed by a 5. The array contains the same number of 4 and 5, and every 4 has a number after it that is not a 4. In this version, 5 may appear anywhere in the original array.
  - b. Implement a method, `testAdjacent()` that takes a two-dimensional `int` array as a parameter, and for each row tests the `makeAdjacent()` method and displays the result.
  - c. Implement the main method that declares a two-dimensional array of integers (sample data shown in the output below) and calls the `testAdjacent()` method to display the output below.

#### Sample Run:

```
Original: [5, 4, 9, 4, 9, 5] Result: [4, 5, 9, 4, 5, 9]
Original: [4, 2, 4, 5, 5] Result: [4, 5, 2, 4, 5]
Original: [5, 4, 5, 4, 1] Result: [4, 5, 4, 5, 1]
Original: [5, 4, 1] Result: [4, 5, 1]
Original: [1, 1, 1] Result: [1, 1, 1]
Original: [4, 5] Result: [4, 5]
```

## Part 2:

1. Implement a class, `Page`, that is the virtual equivalent of a page in a notebook. It has only one private attribute, `contents`, a character array. You should implement the following:
  - **Constructor**: that takes array size as a parameter and initializes content array to its default values ('-' character).
  - **writePage ()** : takes a `String` as a parameter. Begins to write the given string from beginning of the `Page`. Stop writing string if:
    - `contents` array length is exceeded.
    - parameter string's length is exceeded.
  - **toString** method
2. Implement a class, `Notebook`, that simulates a real-life notebook. A `Notebook` has 2 private attributes, a fixed number of pages and an index to keep track of current page.
  - `pages (Page[])`
  - `currentPageIndex (int)`

The `Notebook` class will also include the following methods:

- **Constructor**: Takes page count and page size as parameters and initializes class attributes accordingly.
  - **writeNote ()** : Takes a parameter string and writes it to the **current Page**.
  - **printNotebook ()** : Prints notebook contents to console.
3. Implement the application, `Diary`, that does the following:
    - Input the page count and page size as a parameter from user and create a `Notebook` object.
    - Display the menu shown, which prompts the user with the following options:
      - Add a note to the `Diary`.
      - Print notebook.
      - Exit program.

Review the sample output below, make note of special situations.

**Sample output** *(User input is shown in green)*

Enter number of pages: 5

Enter page size: 20

-----  
What do you want to do?

- (1) Take a note
- (2) Browse notebook
- (3) Exit

Your choice: 1

Enter note: Dear Diary - Here is my first entry!

-----  
What do you want to do?

- (1) Take a note
- (2) Browse notebook
- (3) Exit

Your choice: 1

Enter note: Spring is here, waiting for summer.

-----  
What do you want to do?

- (1) Take a note
- (2) Browse notebook
- (3) Exit

Your choice: 1

Enter note: I hope that my notes will fit into my notebook, I have a lot to say!

-----  
What do you want to do?

- (1) Take a note
- (2) Browse notebook
- (3) Exit

Your choice: 2

Here's your Notebook:

- 0. Dear Diary - Here is
- 1. Spring is here, wait
- 2. I hope that my notes
- 3. -----
- 4. -----

-----  
What do you want to do?

- (1) Take a note
- (2) Browse notebook
- (3) Exit

Your choice: 1

Enter note: Oops, too much!

-----  
What do you want to do?

- (1) Take a note
- (2) Browse notebook
- (3) Exit

Your choice: 1

Enter note: One more thing....

-----  
What do you want to do?

- (1) Take a note
- (2) Browse notebook
- (3) Exit

Your choice: 2

Here's your Notebook:

- 0. Dear Diary - Here is
- 1. Spring is here, wait
- 2. I hope that my notes
- 3. Oops, too much!----
- 4. One more thing....--

-----  
What do you want to do?

- (1) Take a note
- (2) Browse notebook
- (3) Exit

Your choice: 1

Enter note: Here's my last entry for today  
Notebook full! No pages remaining!

-----  
What do you want to do?

- (1) Take a note
- (2) Browse notebook
- (3) Exit

Your choice: 2

Here's your Notebook:

- 0. Dear Diary - Here is
- 1. Spring is here, wait
- 2. I hope that my notes
- 3. Oops, too much!----
- 4. One more thing....--

-----  
What do you want to do?

- (1) Take a note
- (2) Browse notebook
- (3) Exit

Your choice: 3

Goodbye...