HW 06: ArrayBuilder

Release date: Friday, October 7th, 2016
Due date: Friday, October 14th, 2016 – 11:59 pm

Goals

- Learn to create and transverse 2-dimensional arrays

Description

The goal of this homework is to build an array containing alphabet letters in a special order. The program will start by asking a user for a base letter in the alphabet (it could be any letter, either capitalized or in lower case) and the dimensions of the array: number of rows (n) and number of columns (m). Then, it will populate an array nxm that has the base letter in the leftmost corner (0x0 position) and fills the rest of the array with additional letters following a specific order:

- Each letter should be followed by its consequent letter in the alphabet in both vertical and horizontal directions.
- Horizontally, the order goes from left to right, following the array indexes.
- Vertically, the order goes from top to bottom, following the array indexes.

For example, if the base letter is A and the array is of 5x5, the resulting array should look like this.

A	B	C	D	E
B	C	D	E	F
C	D	E	F	G
D	E	F	G	H
E	F	G	H	I

Where A is followed by B both horizontally (going right) and vertically (going down), B is followed by C, and so on. Once we reach Z, the next letter should be A again, following a circular manner.

If the base letter is y and the array is of 4x6, the resulting array should look like this:

y	z	a	b
z	a	b	c
a	b	c	d
b	c	d	e
c	d	e	f
d	e	f	g

The ArrayBuilder class

Model this problem in a class called `ArrayBuilder.` To create a *new* instance of an array builder you will need the base letter and the dimensions of the array. The prototype of the constructor is, thus:

```
public ArrayBuilder(char baseLetter, int n, int m)
```

The constructor should take care of initializing the variables needed to represent this problem: the 2-D char array (name it `letterArray`) and the base letter.

Next, you will create a method:

```
public void build()
```

This method will build the array using the specifications explained in the problem description.

Additionally, you will create a method:

```
public char[][] getLetterArray()
```

to return **a copy** of the array that you have built. Keep in mind that returning the reference to your instance variable and returning a copy of the array are different things.

Finally, create a helper method:

```
public void printLetterArray()
```

to visualize the contents of your array and test the correctness of your program. You are free to choose the way to print this. Use a simple main method to make **several different** calls to your `ArrayBuilder` and observe the output of your program after a call to `build()`. You are also encouraged to write your own test cases.

Rubric

- 10% Constructor
- 10% getLetterArray
- 80% `build`

Submission

Submit all your files to Vocareum by following these instructions. Keep in mind that there is a limit of 20 submissions and only your last submission will be considered.

Good Luck