Homework 5

Question 1 (6 pt.)

Modify the implementation of class `Vector` in such a way that not only it grows its capacity when needed, but also shrinks it as its utilization falls under a certain threshold. Create a directory named `q1`, and follow these steps:

- Copy file `Vector.java` written in class.

- Rename function `Grow()` to `Resize()`. The new version of this function takes an argument of type `int` called `new_size`. Instead of simply doubling the capacity of the vector, it sets the new capacity to the value passed in `new_size`, which is always expected to be equal or greater than field `length`.

- Modify function `Insert()` in a way that it calls the new function `Resize()`, instead of `Grow()`, and in a way that its behavior remains the same as before. This function should continue to print a message showing the new vector's capacity.

- Modify function `Remove()` in a way that it reduces the vector's capacity in half when its utilization is less than 50% of the allocated capacity, and when the resulting capacity would be at least 1 element.

- Write a main program in a separate file named `Test.java` in which you enter 5 elements and extract 4. While running the program, you should observe that the vector grows twice, and later shrinks twice, too.

Pack directory `q1` in a file named `q1.zip` and upload it on Canvas. Your code should compile and execute without errors.

Question 2 (4 pt.)

Create a directory named `q2`, and within it, a file named `Test.java`. This program should declare and initialize an array of 6 integer arbitrary values. The program should then enter an infinite loop in which the following actions occur:

- The program asks the user to enter an integer value, representing an index in the array.
- The program reads the value in the array at the given index. If the index is out of bounds, the array access triggers an exception of type `ArrayIndexOutOfBoundsException`. The program should catch this exception and prevent it from immediately terminating execution.
- If the value in the array was read successfully, the program prints it, and repeats the process. If the inserted index was out of bounds, the program prints a user-friendly message indicating this fact, and finishes execution.

Pack directory `q2` in a file named `q2.zip` and upload it on Canvas. Your code should compile and execute without errors.