COP 3530 Data Structure and Algorithm Analysis
Homework 2

Feng-Hao Liu

In this assignment, you are going to complete the program as specified in "nDVector.cpp". You need to turn in **a single cpp** file via BlackBoard.

For this assignment, you are going to implement a class called nDVector, which stores a n-dimensional point of an arbitrary dimension. In particular, the class has a private integer pointer that points to an array, and an integer size that stores the size of the array. You are going to play with the array and the size.

- **Task 1:** Complete the constructors and the destructor. You can write whatever default constructor of your own design. For the user-specified constructor that takes input an integer (say c), you need to initialize the object with size as the input c, and create an array of that size (i.e. c).
- **Task 2:** Implement several public interface that allows others to read/set the size, and an element in the array. The functionalities are pretty self-explained from their names and input/output types.
- **Task 3:** Write the append and remove function. The append function should "append" the input to the end of the array, and the remove function should remove the p[i] from the array. For example, if your current array is $[1, 2, 3, 4, 5]$, then append(6) should make the array $[1, 2, 3, 4, 5, 6]$. From the new array, if you call remove(3), it should remove p[3], resulting in $[1, 2, 3, 5, 6]$. (Recall that the array starts at 0).
- **Task 4:** Overload the operators as specified in the code. (Remember to do cin and cout).
- **Task 5:** In your main function, write several test cases that test your implementations. The style is flexible.