

## CSE 330 Homework Assignment 1 , Spring 2018

Instructor: Kerstin Voigt

Complete the ADT Vector implementation from Lab2 and add the following two functions to the Vector class:

1. Member function `void erase(int k) {...}` which removes the element at index `k`.
2. Member function `void insert(int k, T x) {...}` which inserts the new element `x` at index `k`. (The element that is at index `k` prior to insertion, as well as all other elements at higher-numbered indices are to move one position to the right.)

Details of the processes behind both functions are depicted below.

**Erase:** Erase the element at index 3 in vector `[2][4][6][8][10][12][14][16]`

Step-by-step ...

```
    0 1 2 3 4 5 6 7
... [2][4][6][8][10][12][14][16]

... [2][4][6][10][10][12][14][16]

... [2][4][6][10][12][12][14][16]

... [2][4][6][10][12][14][14][16]

... [2][4][6][10][12][14][16][16]

... [2][4][6][10][12][14][16][16]

... [2][4][6][10][12][14][16]  DONE!
```

Look at the example and realize the kinds of shifts (really, “copies”) that will have the desired effect. Also notice the duplicate element at the end and how it is removed in order to produce the final result.

Your function should also handle the special case where the removal index is the index of the element at the highest index ... what will be the simplest way to accomplish this element’s removal? (Hint: this function already exists).

**Insert:** Insert new element 7 at index 3 in vector [2][4][6][8][10][12][14][16]

Step-by-step ...

```
      0  1  2  3  4  5  6  7
... [2][4][6][8][10][12][14][16]

... [2][4][6][8][10][12][14][16][16]

... [2][4][6][8][10][12][14][14][16]

... [2][4][6][8][10][12][12][14][16]

... [2][4][6][8][10][10][12][14][16]

... [2][4][6][8][8][10][12][14][16]

... [2][4][6][7][8][10][12][14][16]  DONE!
```

Look at the example and realize the kinds of shifts (really, “copies”) that will have the desired effect. Also notice how the process starts out with the addition of a copy of the last element at the right of end of the vector.

Your function should also handle the special case where the insertion index is the index that is one more than the highest index pre-insertion ... what will be the simplest way to accomplish insertion into this index? (Hint: this function already exists)

Test your new member functions with a simple int main() function that defines a Vector of integer values, fills it up with 10 values of your choice, and performs erase and insert functions as follows:

1. Erase the element at the highest index.
2. Erase the element at index 3.
3. Insert integer 777 at the position 10.
4. Insert integer 777 at 5.

**Submit on Thursday, April 19, 2018, at the beginning of the lecture:** (1) a hardcopy of your file Vector.h with highlighting of your member functions erase and insert, (2) a hardcopy of your file VectorHw1Main.cpp with the int main() you wrote for testing 1.-4., and (3) a typescript that shows your compiling and running the program . In case you do not achieve a running program, submit a typescript that shows your attempt to compile and the compiler errors you get.