CS 2401 Spring 2022

**Lab 4: Dynamic Arrays and The Big 3**

**Due: Friday 2/11 at 11:59 PM – Hunter Stout**

Type your answers on this sheet making sure to list the question numbers.

1. It’s preforming the code correctly for the most part, although when it copies n1 to n2 its messing up the used value so its only performing the remove\_last function so the last 14 is staying in the array.
2. Yes this is quite the problem because its not performing correctly. The second array should consist of 2, 4, 6, 5, 10, 15. Although there is a 14 after the 15.
3. The used variable is messing up when the class copies. When I comment out the copy in main it adds them properly, and gives me an empty array message 4 times, although when I keep the copy function it only removes 3 values. The error must be since we are missing 2 of the big 3.
4. Now its different than before. Its just straight copying the array. Outputs 1 & 2 are the same and it is completely disregarding the remove function with this operator.
5. Yes, it is different than what I saw before. It will run the remove function, but I think its now fighting the assignment operator in the code causing it to not even change the second array.
6. I think the assignment operator is not allowing it to change for some reason.
7. 0xf46f48, 0xf470e0, 0xf47278, 0xf47410, 0xf475a8.
8. They are 408 bits apart from each other. So, I would say since there are 8 bits in a byte, 51 bytes between each outputted value. The addresses converted are: 16019272, 16019680, 16020088, 16020496, 16020904.
9. All 5 addresses are the same now “0xe56f48”. Then it outputs my deconstructor cout after each address saying “array memory cleared”.
10. The byte count is back down to 60, it was 2060 before the deconstructor was added.
11. Adding the deconstructor cleared the dynamic array memory everytime it left scope, therefore every time in main when it leaves the “for loop” and calls the cout displaying reveal address it deconstructs the array. Since the array is deconstructed each time and the byte count is reset. The address of the array is reset instead of pilling up memory in the program.