All code is found in lab07.sln- seventh try at the code, unfortunately VScode crashed a few times and I was lazy to change the name of the document

Task 1: what is trying to be achieved, what are the strengths and weaknesses, and how will it perform?

It is trying to add items to the array by starting from the beginning of the array and then finding the location where the item should be added. The remove function also starts from the beginning and then finds the item to be removed. This is the base class and we expect that adding items and removing items that are closer to the end of the array will take the most number of operations to achieve using this approach. While adding/removing items that are closer to beginning will take the least number of operations.

Task 2: what is trying to be achieved, what are the strengths and weaknesses, and how will it perform?

In Task 2, we are trying to see if starting at the end of the array will help reduce the total number of operations to add and remove items from it. For example, to add an item we will start at the end of the array, and then move to the item at the end of the array. We believe that this should perform better than the base class approach of adding and removing items, since we start at the end instead of the beginning of the array, hence making it costs the same operations to add/remove items closer to the end of the array.

Task 3: what is trying to be achieved, what are the strengths and weaknesses, and how will it perform?

In Task 3, we are trying to reduce the number of moves when inserting an item into the ordered list by inserting an item halfway between any two items in the array where it belongs. This way, items are only moved if the inserting item sits between two items that are in contiguous locations. The RemoveItem method doesn't move any items in the array, and instead it makes that spot null. There should be less operations which is a strength, but it may potentially take longer for each operation to be performed, which could be a weakness.

Task 4

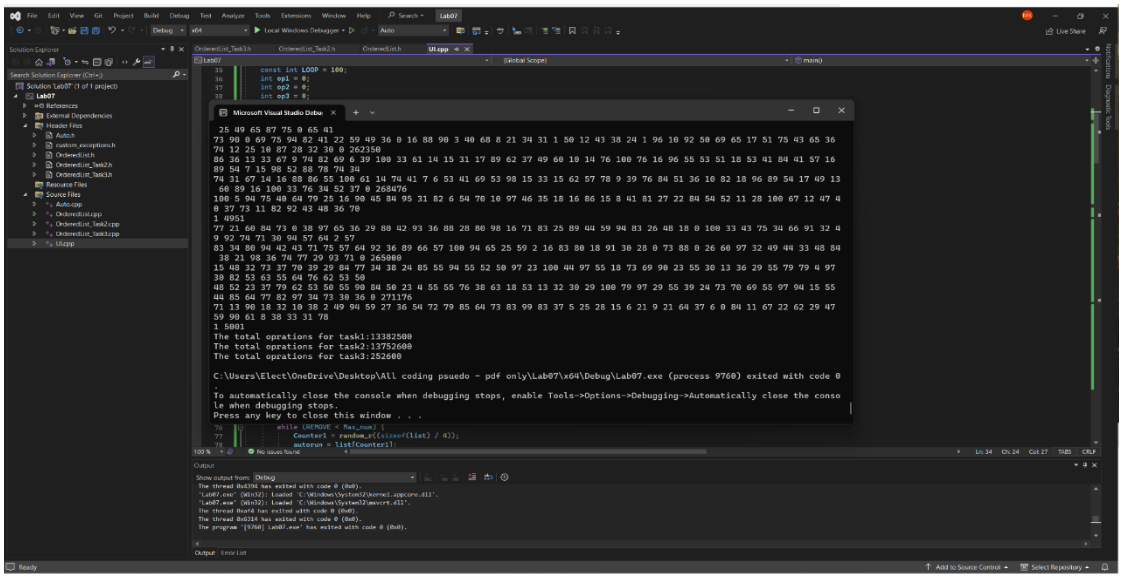
Describe results when running it 100 times (base test case of array size 25):

The results of the Test were 3535000 for task 1 , 3688850 for task 2 , 126350 for task 3. For task1 - individually it matches the expected result, for task 2 - it matches the expected result, and for task 3 the base code is faulty and as such is not equal to the expected value.

Describe results when array size is increased to 50 items and ran 100 times:
For task 1 the results for operations for task 1: 13382500, task 2: 13752600, task 3: 252600 For task 1 and task 2 the results are at the expected values. For part three though the remove function is faulty and thus does not meet the expected value.

Describe results when array size is decreased to 10 items and ran 100 times:
For part 3 the results for task 1: 656500 , task 2: 710600 , task 3: 50600. For task 1 and task 2 are meeting the expected value, while task 3 is not as the remove function is faulty thus not meeting the required value.

Screenshots


File Edit View Git Project Build Debug Test Analyze Tools Extensions Window Help Search Lab07
Debug x64 Local Windows Debugger Auto
Live Share
Solution Explorer
Search Solution Explorer (Ctrl+;)
Solution 'Lab07' (1 of 1 project)
Lab07
References
External Dependencies
Header Files
Auto.h
custom_exceptions.h
OrderedList.h
OrderedList_Task2.h
OrderedList_Task3.h
Resource Files
Source Files
Auto.cpp
OrderedList.cpp
OrderedList_Task2.cpp
OrderedList_Task3.cpp
UI.cpp
OrderedList_Task3.h OrderedList_Task2.h OrderedList.h UI.cpp
Lab07 (Global Scope) main()
Microsoft Visual Studio Debu
100 % No issues found
Ln: 34 Ch: 24 Col: 27 TABS CRLF
Output
Show output from: Debug
Output Error List
Ready
Add to Source Control Select Repository

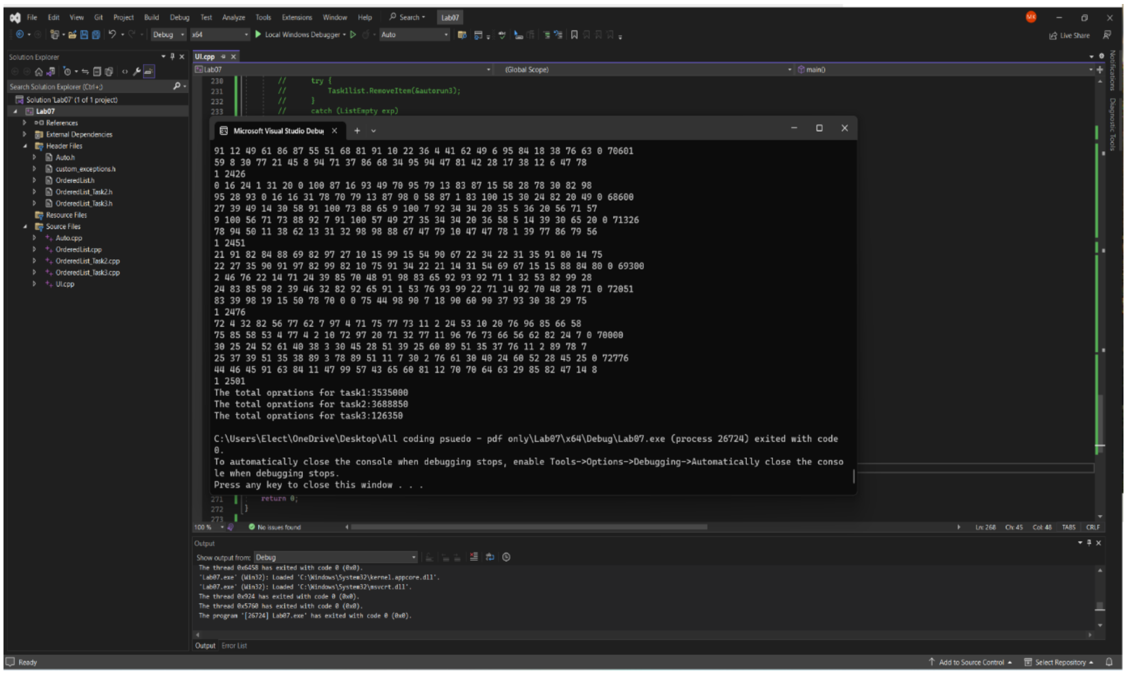
```
    const int LOOP = 100;
    int op1 = 0;
    int op2 = 0;
    int op3 = 0;
```

```
 25 49 65 87 75 0 65 41
73 90 0 69 75 94 82 41 22 59 49 36 0 16 88 90 3 40 68 8 21 34 31 1 50 12 43 38 24 1 96 10 92 50 69 65 17 51 75 43 65 36
74 12 25 10 87 28 32 30 0 262350
86 36 13 33 67 9 74 82 69 6 39 100 33 61 14 15 31 17 89 62 37 49 60 10 14 76 100 76 16 96 55 53 51 18 53 41 84 41 57 16
89 54 7 15 98 52 88 78 74 34
74 31 67 14 16 88 86 55 100 61 14 74 41 7 6 53 41 69 53 98 15 33 15 62 57 78 9 39 76 84 51 36 10 82 18 96 89 54 17 49 13
 60 89 16 100 33 76 34 52 37 0 268476
100 5 94 75 40 64 79 25 16 90 45 84 95 31 82 6 54 70 10 97 46 35 18 16 86 15 8 41 81 27 22 84 54 52 11 28 100 67 12 47 4
0 37 73 11 82 92 43 48 36 70
1 4951
77 21 60 84 73 0 38 97 65 36 29 80 42 93 36 88 28 80 98 16 71 83 25 89 44 59 94 83 26 48 18 0 100 33 43 75 34 66 91 32 4
9 92 74 71 30 94 57 64 2 57
83 34 80 94 42 43 71 75 57 64 92 36 89 66 57 100 94 65 25 59 2 16 83 80 18 91 30 28 0 73 88 0 26 60 97 32 49 44 33 48 84
 38 21 98 36 74 77 29 93 71 0 265000
15 48 32 73 37 70 39 29 84 77 34 38 24 85 55 94 55 52 50 97 23 100 44 97 55 18 73 69 90 23 55 30 13 36 29 55 79 79 4 97
30 82 53 63 55 64 76 62 53 50
48 52 23 37 79 62 53 50 55 90 84 50 23 4 55 55 76 38 63 18 53 13 32 30 29 100 79 97 29 55 39 24 73 70 69 55 97 94 15 55
44 85 64 77 82 97 34 73 30 36 0 271176
71 13 90 18 32 10 38 2 49 94 59 27 36 54 72 79 85 64 73 83 99 83 37 5 25 28 15 6 21 9 21 64 37 6 0 84 11 67 22 62 29 47
59 90 61 8 38 33 31 78
1 5001
The total oprations for task1:13382500
The total oprations for task2:13752600
The total oprations for task3:252600

C:\Users\Elect\OneDrive\Desktop\All coding psuedo - pdf only\Lab07\x64\Debug\Lab07.exe (process 9760) exited with code 0
.
To automatically close the console when debugging stops, enable Tools->Options->Debugging->Automatically close the conso
le when debugging stops.
Press any key to close this window . . .
```

```
        while (REMOVE < Max_num) {
            Counter1 = random_r((sizeof(list) / 4));
            autorun = list[Counter1];
```

```
The thread 0x4394 has exited with code 0 (0x0).
'Lab07.exe' (Win32): Loaded 'C:\Windows\System32\kernel.appcore.dll'.
'Lab07.exe' (Win32): Loaded 'C:\Windows\System32\msvcrt.dll'.
The thread 0xaf4 has exited with code 0 (0x0).
The thread 0x6314 has exited with code 0 (0x0).
The program '[9760] Lab07.exe' has exited with code 0 (0x0).
```



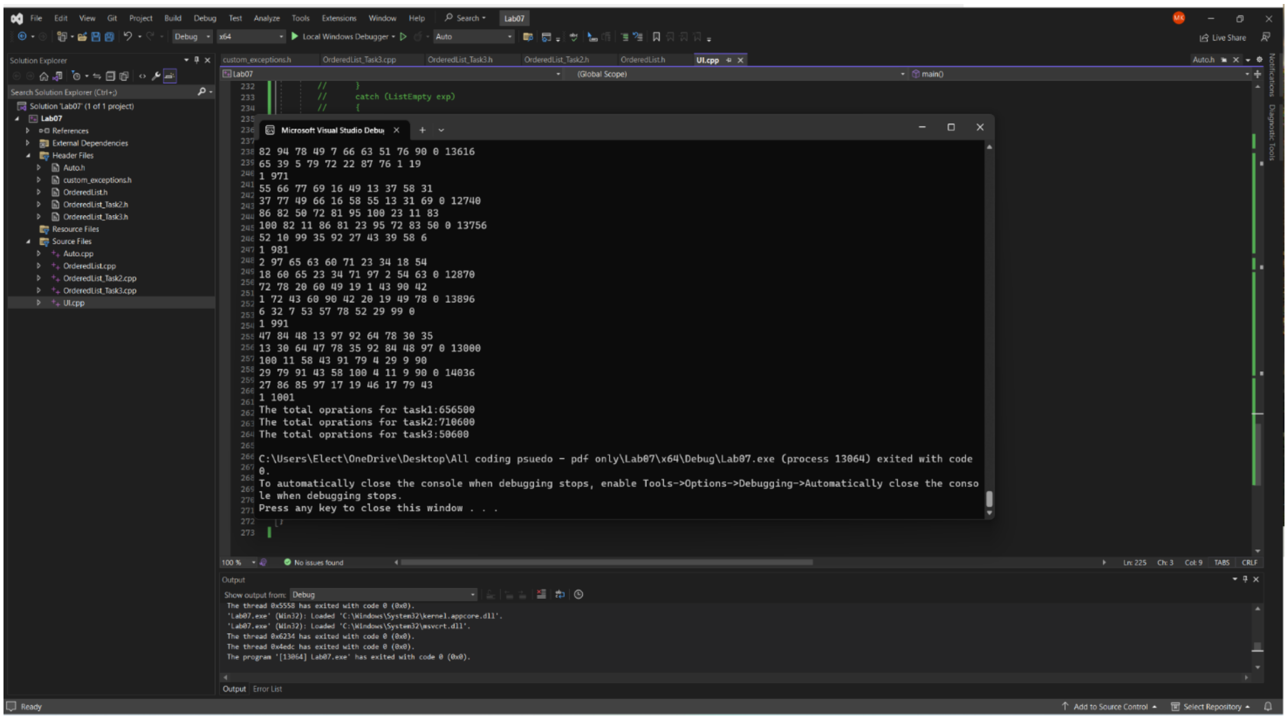
```
    //      try {
    //          Task1list.RemoveItem(&autorun3);
    //      }
    //      catch (ListEmpty exp)
```

```
91 12 49 61 86 87 55 51 68 81 91 10 22 36 4 41 62 49 6 95 84 18 38 76 63 0 70601
59 8 30 77 21 45 8 94 71 37 86 68 34 95 94 47 81 42 28 17 38 12 6 47 78
1 2426
0 16 24 1 31 20 0 100 87 16 93 49 70 95 79 13 83 87 15 58 28 78 30 82 98
95 28 93 0 16 16 31 78 70 79 13 87 98 0 58 87 1 83 100 15 30 24 82 20 49 0 68600
27 39 49 14 30 58 91 100 73 88 65 9 100 7 92 34 34 20 35 5 36 20 56 71 57
9 100 56 71 73 88 92 7 91 100 57 49 27 35 34 34 20 36 58 5 14 39 30 65 20 0 71326
78 94 50 11 38 62 13 31 32 98 98 88 67 47 79 10 47 47 78 1 39 77 86 79 56
1 2451
21 91 82 84 88 69 82 97 27 10 15 99 15 54 90 67 22 34 22 31 35 91 80 14 75
22 27 35 90 91 97 82 99 82 10 75 91 34 22 21 14 31 54 69 67 15 15 88 84 80 0 69300
2 46 76 22 14 71 24 39 85 70 48 91 98 83 65 92 93 92 71 1 32 53 82 99 28
24 83 85 98 2 39 46 32 82 92 65 91 1 53 76 93 99 22 71 14 92 70 48 28 71 0 72051
83 39 98 19 15 50 78 70 0 0 75 44 98 90 7 18 90 60 90 37 93 30 38 29 75
1 2476
72 4 32 82 56 77 62 7 97 4 71 75 77 73 11 2 24 53 10 20 76 96 85 66 58
75 85 58 53 4 77 4 2 10 72 97 20 71 32 77 11 96 76 73 66 56 62 82 24 7 0 70000
30 25 24 52 61 40 38 3 30 45 28 51 39 25 60 89 51 35 37 76 11 2 89 78 7
25 37 39 51 35 38 89 3 78 89 51 11 7 30 2 76 61 30 40 24 60 52 28 45 25 0 72776
44 46 45 91 63 84 11 47 99 57 43 65 60 81 12 70 70 64 63 29 85 82 47 14 8
1 2501
The total oprations for task1:3535000
The total oprations for task2:3688850
The total oprations for task3:126350

C:\Users\Elect\OneDrive\Desktop\All coding psuedo - pdf only\Lab07\x64\Debug\Lab07.exe (process 26724) exited with code
0.
To automatically close the console when debugging stops, enable Tools->Options->Debugging->Automatically close the conso
le when debugging stops.
Press any key to close this window . . .
```

```
    return 0;
}
```

```
The thread 0x6458 has exited with code 0 (0x0).
'Lab07.exe' (Win32): Loaded 'C:\Windows\System32\kernel.appcore.dll'.
'Lab07.exe' (Win32): Loaded 'C:\Windows\System32\msvcrt.dll'.
The thread 0x924 has exited with code 0 (0x0).
The thread 0x5760 has exited with code 0 (0x0).
The program '[26724] Lab07.exe' has exited with code 0 (0x0).
```

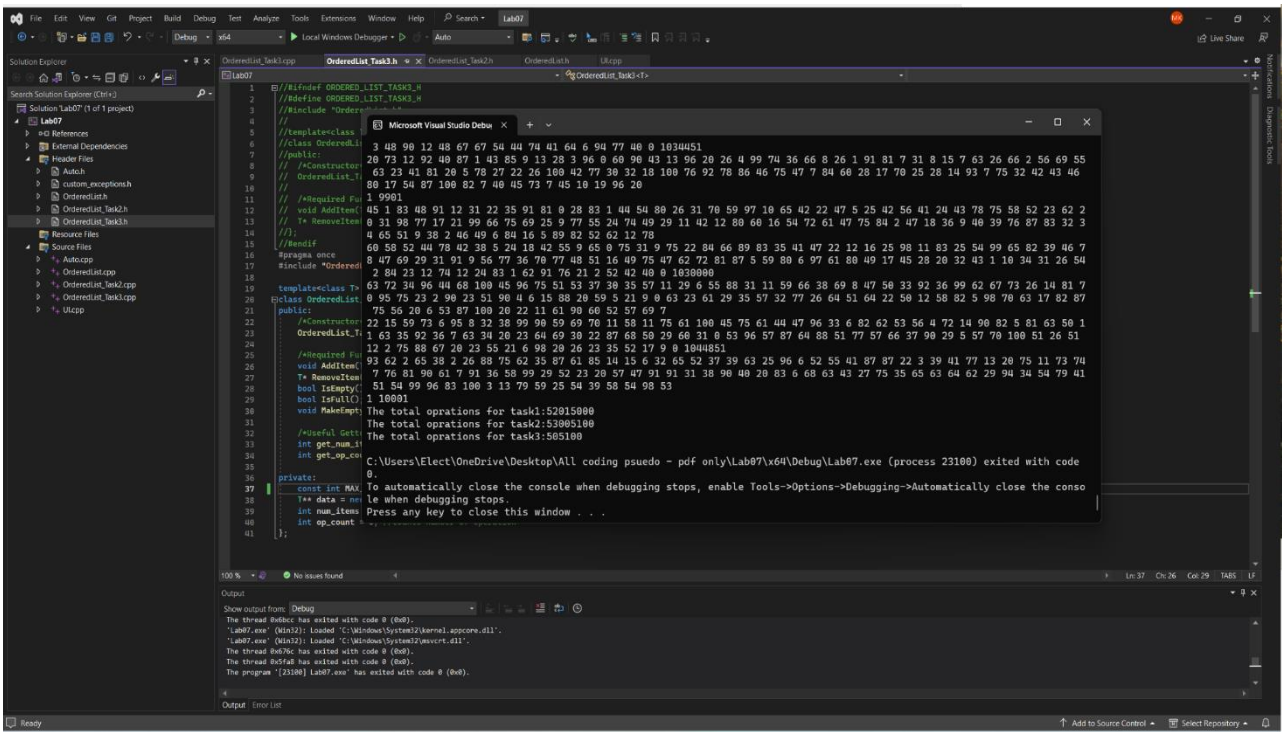


```
//      }
//      catch (ListEmpty exp)
//      {
```

```
82 94 78 49 7 66 63 51 76 90 0 13616
65 39 5 79 72 22 87 76 1 19
1 971
55 66 77 69 16 49 13 37 58 31
37 77 49 66 16 58 55 13 31 69 0 12740
86 82 50 72 81 95 100 23 11 83
100 82 11 86 81 23 95 72 83 50 0 13756
52 10 99 35 92 27 43 39 58 6
1 981
2 97 65 63 60 71 23 34 18 54
18 60 65 23 34 71 97 2 54 63 0 12870
72 78 20 60 49 19 1 43 90 42
1 72 43 60 90 42 20 19 49 78 0 13896
6 32 7 53 57 78 52 29 99 0
1 991
47 84 48 13 97 92 64 78 30 35
13 30 64 47 78 35 92 84 48 97 0 13000
100 11 58 43 91 79 4 29 9 90
29 79 91 43 58 100 4 11 9 90 0 14036
27 86 85 97 17 19 46 17 79 43
1 1001
The total oprations for task1:656500
The total oprations for task2:710600
The total oprations for task3:50600

C:\Users\Elect\OneDrive\Desktop\All coding psuedo - pdf only\Lab07\x64\Debug\Lab07.exe (process 13064) exited with code
0.
To automatically close the console when debugging stops, enable Tools->Options->Debugging->Automatically close the conso
le when debugging stops.
Press any key to close this window . . .
```

```
The thread 0x5558 has exited with code 0 (0x0).
'Lab07.exe' (Win32): Loaded 'C:\Windows\System32\kernel.appcore.dll'.
'Lab07.exe' (Win32): Loaded 'C:\Windows\System32\msvcrt.dll'.
The thread 0x6234 has exited with code 0 (0x0).
The thread 0x4edc has exited with code 0 (0x0).
The program '[13064] Lab07.exe' has exited with code 0 (0x0).
```



```
//#ifndef ORDERED_LIST_TASK3_H
//#define ORDERED_LIST_TASK3_H
//#include "Ordere
//
//template<class
//class OrderedLi
//public:
//  /*Constructor
//  OrderedList_T
//
//  /*Required Fu
//  void AddItem(
//  T* RemoveItem
//};
//#endif
#pragma once
#include "Ordered

template<class T>
class OrderedList_
public:
    /*Constructor
    OrderedList_T

    /*Required Fu
    void AddItem(
    T* RemoveItem
    bool IsEmpty(
    bool IsFull()
    void MakeEmpt

    /*Useful Gett
    int get_num_i
    int get_op_co

private:
    const int MAX
    T** data = ne
    int num_items
    int op_count
};
```

```
 3 48 90 12 48 67 67 54 44 74 41 64 6 94 77 40 0 1034451
20 73 12 92 40 87 1 43 85 9 13 28 3 96 0 60 90 43 13 96 20 26 4 99 74 36 66 8 26 1 91 81 7 31 8 15 7 63 26 66 2 56 69 55
 63 23 41 81 20 5 78 27 22 26 100 42 77 30 32 18 100 76 92 78 86 46 75 47 7 84 60 28 17 70 25 28 14 93 7 75 32 42 43 46
80 17 54 87 100 82 7 40 45 73 7 45 10 19 96 20
1 9901
45 1 83 48 91 12 31 22 35 91 81 0 28 83 1 44 54 80 26 31 70 59 97 10 65 42 22 47 5 25 42 56 41 24 43 78 75 58 52 23 62 2
0 31 98 77 17 21 99 66 75 69 25 9 77 55 24 74 49 29 11 42 12 80 60 16 54 72 61 47 75 84 2 47 18 36 9 40 39 76 87 83 32 3
4 65 51 9 38 2 46 49 6 84 16 5 89 82 52 62 12 78
60 58 52 44 78 42 38 5 24 18 42 55 9 65 0 75 31 9 75 22 84 66 89 83 35 41 47 22 12 16 25 98 11 83 25 54 99 65 82 39 46 7
8 47 69 29 31 91 9 56 77 36 70 77 48 51 16 49 75 47 62 72 81 87 5 59 80 6 97 61 80 49 17 45 28 20 32 43 1 10 34 31 26 54
 2 84 23 12 74 12 24 83 1 62 91 76 21 2 52 42 40 0 1030000
63 72 34 96 44 68 100 45 96 75 51 53 37 30 35 57 11 29 6 55 88 31 11 59 66 38 69 8 47 50 33 92 36 99 62 67 73 26 14 81 7
0 95 75 23 2 90 23 51 90 4 6 15 88 20 59 5 21 9 0 63 23 61 29 35 57 32 77 26 64 51 64 22 50 12 58 82 5 98 70 63 17 82 87
 75 56 20 6 53 87 100 20 22 11 61 90 60 52 57 69 7
22 15 59 73 6 95 8 32 38 99 90 59 69 70 11 58 11 75 61 100 45 75 61 44 47 96 33 6 82 62 53 56 4 72 14 90 82 5 81 63 50 1
1 63 35 92 36 7 63 34 20 23 64 69 30 22 87 68 50 29 60 31 0 53 96 57 87 64 88 51 77 57 66 37 90 29 5 57 70 100 51 26 51
12 2 75 88 67 20 23 55 21 6 98 20 26 23 35 52 17 9 0 1044851
93 62 2 65 38 2 26 88 75 62 35 87 61 85 14 15 6 32 65 52 37 39 63 25 96 6 52 55 41 87 87 22 3 39 41 77 13 20 75 11 73 74
 7 76 81 90 61 7 91 36 58 99 29 52 23 20 57 47 91 91 31 38 90 40 20 83 6 68 63 43 27 75 35 65 63 64 62 29 94 34 54 79 41
 51 54 99 96 83 100 3 13 79 59 25 54 39 58 54 98 53
1 10001
The total oprations for task1:52015000
The total oprations for task2:53005100
The total oprations for task3:505100

C:\Users\Elect\OneDrive\Desktop\All coding psuedo - pdf only\Lab07\x64\Debug\Lab07.exe (process 23100) exited with code
0.
To automatically close the console when debugging stops, enable Tools->Options->Debugging->Automatically close the conso
le when debugging stops.
Press any key to close this window . . .
```

```
The thread 0x6bcc has exited with code 0 (0x0).
'Lab07.exe' (Win32): Loaded 'C:\Windows\System32\kernel.appcore.dll'.
'Lab07.exe' (Win32): Loaded 'C:\Windows\System32\msvcrt.dll'.
The thread 0x676c has exited with code 0 (0x0).
The thread 0x5fa8 has exited with code 0 (0x0).
The program '[23100] Lab07.exe' has exited with code 0 (0x0).
```

Discussion of how the program was designed to simplify running and reporting results 100 times. Discuss methods for analyzing the results and why those methods are valid. Discuss other methods you discussed as a group and decided not to use

The program was streamline by using only one while function to run it 100 times that adds up using example# +=# of operations then print the total. Along the way as it was going over each part of the three tasks, it would print the inserted numbers and show the order in which it was deconstructed. From there it then prints out if it ran 25 times and then removed the 25 items. From there it prints out the total amount of time the add and remove operator ran. Another way to potentially look at the results is to store it in a stored variable in the class that stores the total operations over the entire code run. Other methods discussed as a group were to run each task 100 times instead of combined and compartmentalize the code into different compartments to spare on resources. However this was rejected as it creates unnecessary code and increases the files length.