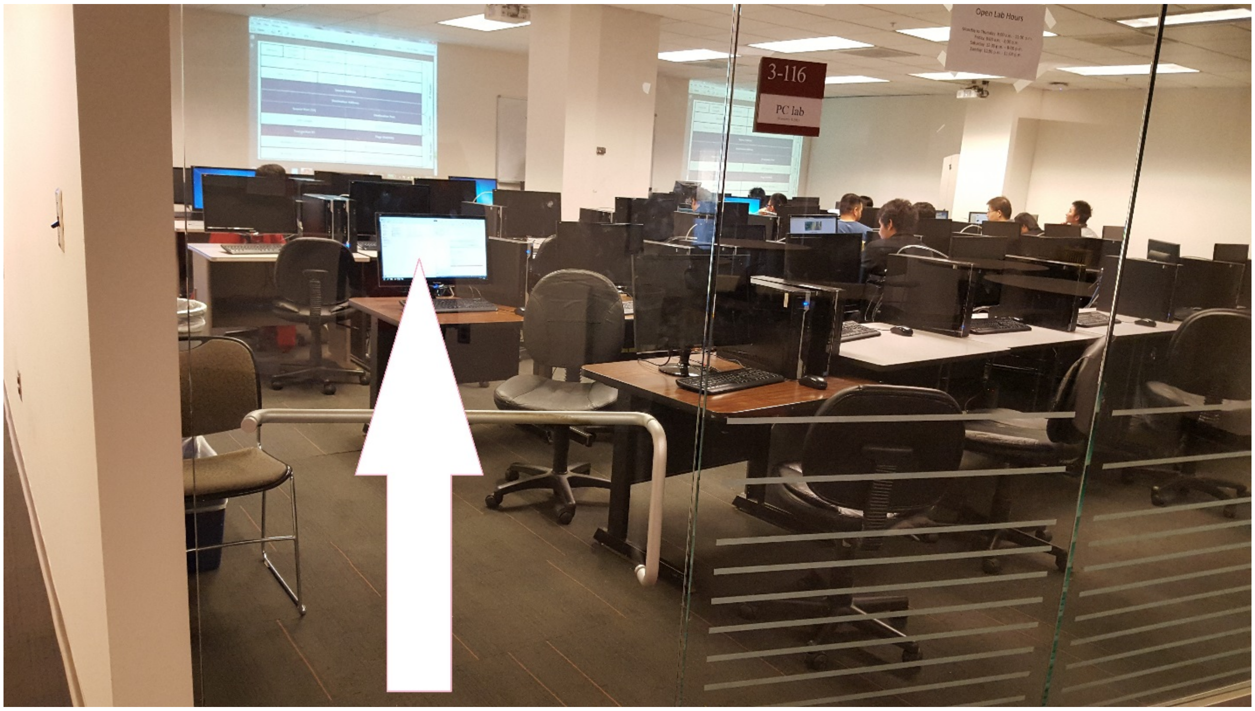
March 11, 2016

Dear Dr. Lee,

How are you? This document will provide instructions to run JaeNwaweHeapifyExchange in Visual Studio 2015 in the LSC 3rd floor 3-116 public lab. I ran it on the second to last row, and the left aisle computer.

The computer is shown below with the computer that is on in the photo with an arrow pointing to it.



I have provided instructions to run Visual Studio 2015 and open this project starting on page 2. The summary of the instructions is that you use a flash drive, and double click the JaeNwaweHeapify solution file. When Visual Studio 2015 opens, select “yes” when it asks you to compile. Then, hit the run button which is green and labeled “Local Windows Debugger”.

Dr. Lee, I am excited about discussing this project with you. I learned a lot, and I will meet with you to answer any questions or help you run Visual Studio 2015 in the public lab.

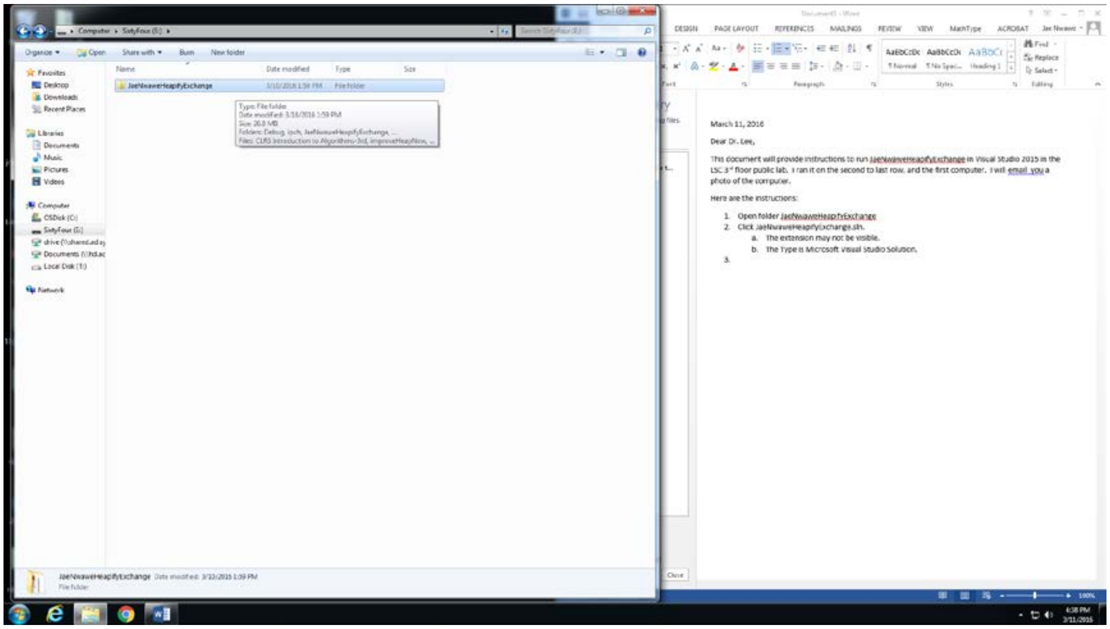
Regards,

Jae Nwawe

P.S. I would like to minimize the time that you have to spend on this since I value your time and know that you are very busy..

Here are the instructions to run this program from a flash drive:

1. Open folder JaeNwaweHeapifyExchange.



Name	Date modified	Type	Size
JaeNwaweHeapifyExchange	3/11/2016 1:59 PM	File folder	

Type: File folder
Date modified: 3/11/2016 1:59 PM
Size: 26.0 MB
Folders: Debug, ipch, JaeNwaweHeapifyExchange, ...
Files: CLRS Introduction to Algorithms-3rd, improveHeapNow, ...

March 11, 2016

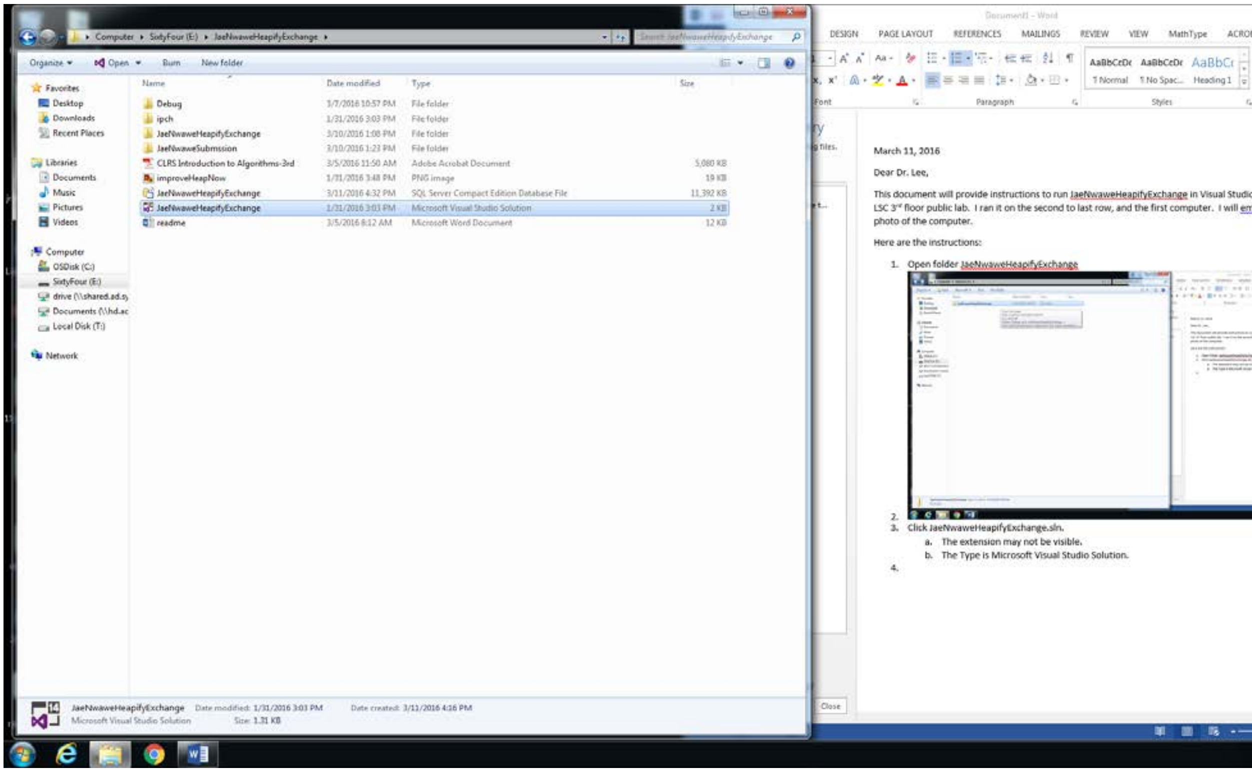
Dear Dr. Lee,

This document will provide instructions to run JaeNwaweHeapifyExchange in Visual Studio 2015 in the LSC 3rd floor public lab. I ran it on the second to last row, and the first computer. I will email you a photo of the computer.

Here are the instructions:

1. Open folder JaeNwaweHeapifyExchange
2. Click JaeNwaweHeapifyExchange.sln.
   a. The extension may not be visible.
   b. The Type is Microsoft Visual Studio Solution.
3.

2. Double click JaeNwaweHeapifyExchange.sln .
   a. The extension may not be visible.
   b. The Type is Microsoft Visual Studio Solution.



Name	Date modified	Type	Size
Debug	1/7/2016 10:57 PM	File folder	
ipch	1/31/2016 3:03 PM	File folder	
JaeNwaweHeapifyExchange	3/10/2016 1:08 PM	File folder	
JaeNwaweSubmission	3/10/2016 1:23 PM	File folder	
CLRS Introduction to Algorithms-3rd	3/5/2016 11:50 AM	Adobe Acrobat Document	5,080 KB
improveHeapNow	1/31/2016 3:48 PM	PNG image	19 KB
JaeNwaweHeapifyExchange	3/11/2016 4:32 PM	SQL Server Compact Edition Database File	11,392 KB
JaeNwaweHeapifyExchange	1/31/2016 3:03 PM	Microsoft Visual Studio Solution	2 KB
readme	3/5/2016 8:12 AM	Microsoft Word Document	12 KB

JaeNwaweHeapifyExchange Date modified: 1/31/2016 3:03 PM Date created: 3/11/2016 4:16 PM
Microsoft Visual Studio Solution Size: 1.31 KB

March 11, 2016

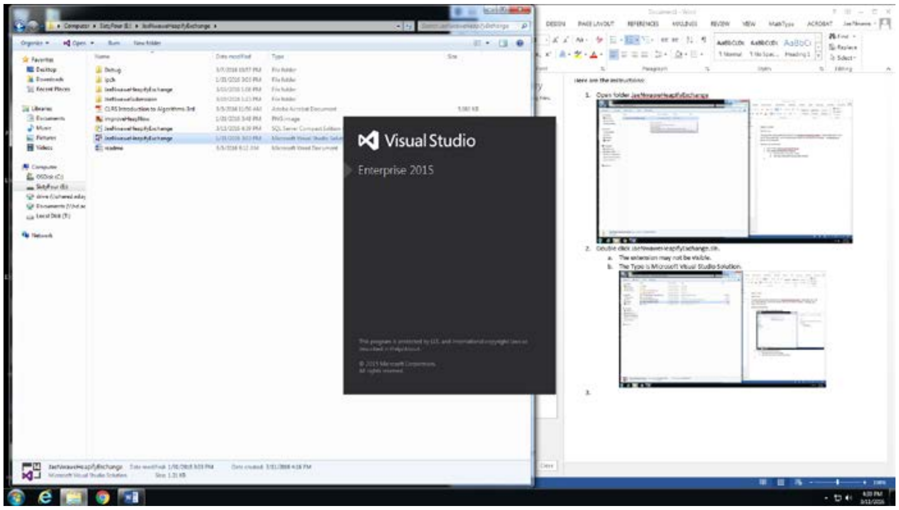
Dear Dr. Lee,

This document will provide instructions to run JaeNwaweHeapifyExchange in Visual Studi
LSC 3rd floor public lab. I ran it on the second to last row, and the first computer. I will em
photo of the computer.

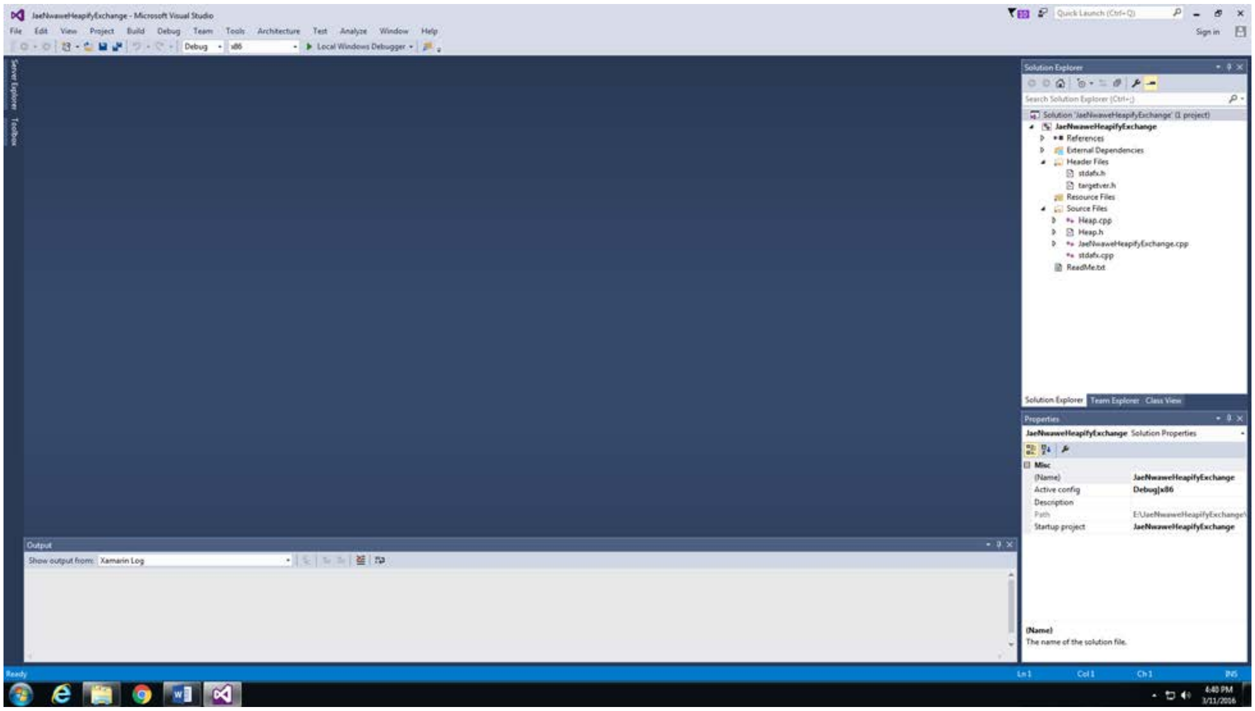
Here are the instructions:

1. Open folder JaeNwaweHeapifyExchange
2.
3. Click JaeNwaweHeapifyExchange.sln.
   a. The extension may not be visible.
   b. The Type is Microsoft Visual Studio Solution.
4.

3. Visual Studio Enterprise 2015 will open and load the solution and project.



4. The project contents will be visible in the “Solution Explorer”.



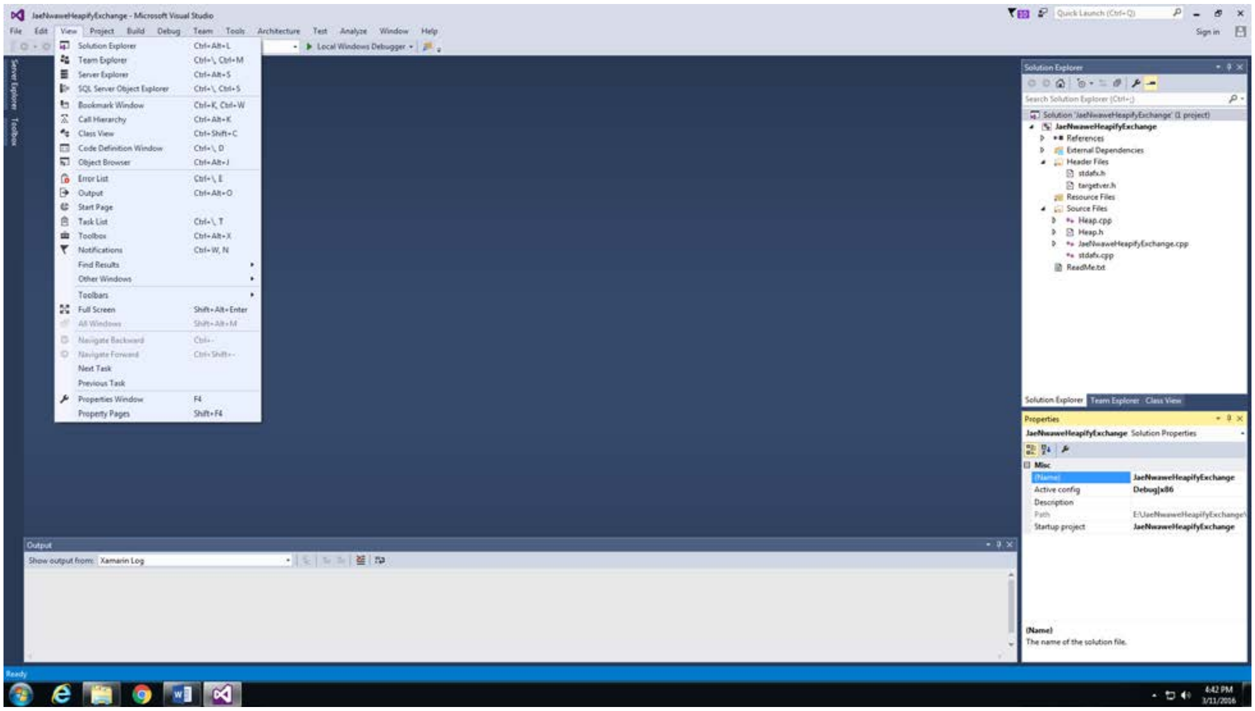
Solution 'JaeNwaweHeapifyExchange' (1 project)
- JaeNwaweHeapifyExchange
  - References
  - External Dependencies
  - Header Files
    - stdafx.h
    - targetver.h
  - Resource Files
  - Source Files
    - Heap.cpp
    - Heap.h
    - JaeNwaweHeapifyExchange.cpp
    - stdafx.cpp
  - ReadMe.txt

JaeNwaweHeapifyExchange Solution Properties

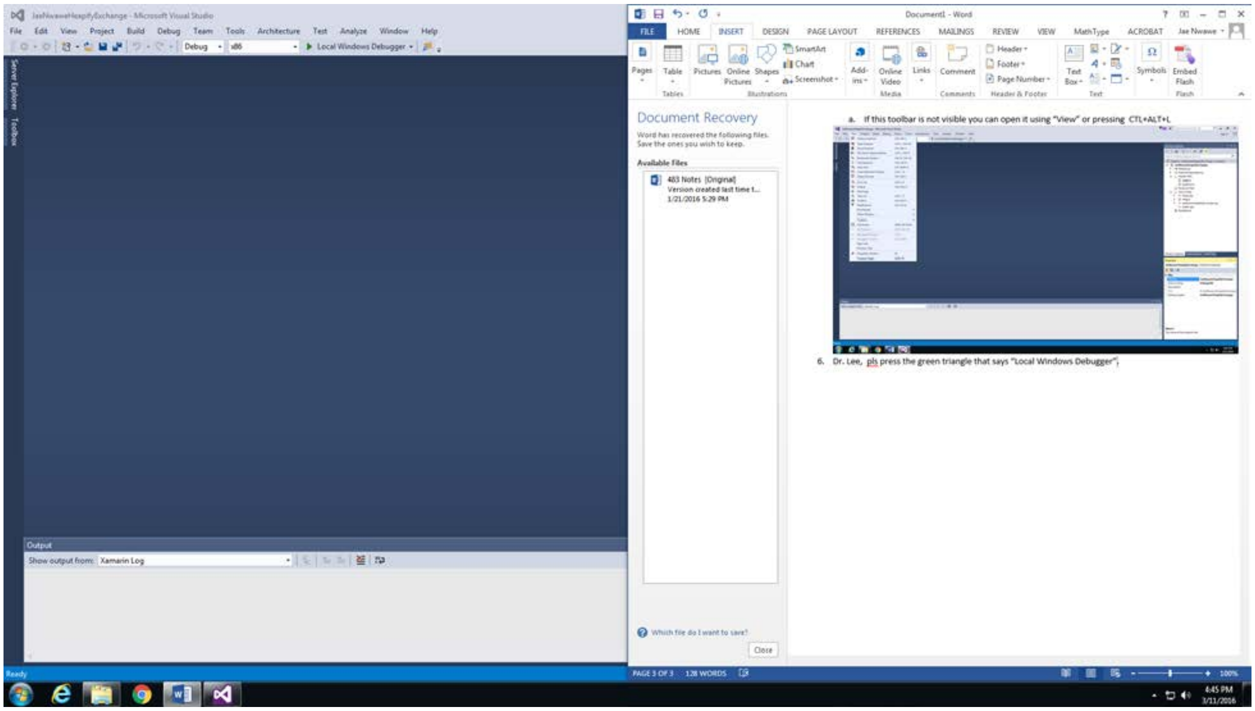
Misc	
(Name)	JaeNwaweHeapifyExchange
Active config	Debug|x86
Description	
Path	E:\JaeNwaweHeapifyExchange\
Startup project	JaeNwaweHeapifyExchange

(Name)
The name of the solution file.

a. If this toolbar is not visible you can open it using “View” or pressing CTL+ALT+L .


JaeNwaweHeapifyExchange - Microsoft Visual Studio
File Edit View Project Build Debug Team Tools Architecture Test Analyze Window Help
Local Windows Debugger
Solution Explorer Ctrl+Alt+L
Team Explorer Ctrl+\, Ctrl+M
Server Explorer Ctrl+Alt+S
SQL Server Object Explorer Ctrl+\, Ctrl+S
Bookmark Window Ctrl+K, Ctrl+W
Call Hierarchy Ctrl+Alt+K
Class View Ctrl+Shift+C
Code Definition Window Ctrl+\, D
Object Browser Ctrl+Alt+J
Error List Ctrl+\, E
Output Ctrl+Alt+O
Start Page
Task List Ctrl+\, T
Toolbox Ctrl+Alt+X
Notifications Ctrl+W, N
Find Results
Other Windows
Toolbars
Full Screen Shift+Alt+Enter
All Windows Shift+Alt+M
Navigate Backward Ctrl+-
Navigate Forward Ctrl+Shift+-
Next Task
Previous Task
Properties Window F4
Property Pages Shift+F4
Solution Explorer
Solution 'JaeNwaweHeapifyExchange' (1 project)
JaeNwaweHeapifyExchange
References
External Dependencies
Header Files
stdafx.h
targetver.h
Resource Files
Source Files
Heap.cpp
Heap.h
JaeNwaweHeapifyExchange.cpp
stdafx.cpp
ReadMe.txt
Properties
(Name) JaeNwaweHeapifyExchange
Active config Debug|x86
Startup project JaeNwaweHeapifyExchange
Output
Show output from: Xamarin Log
Ready
6:42 PM
3/11/2016

5. Please press the green triangle that says “Local Windows Debugger”. It is in the center of the page.

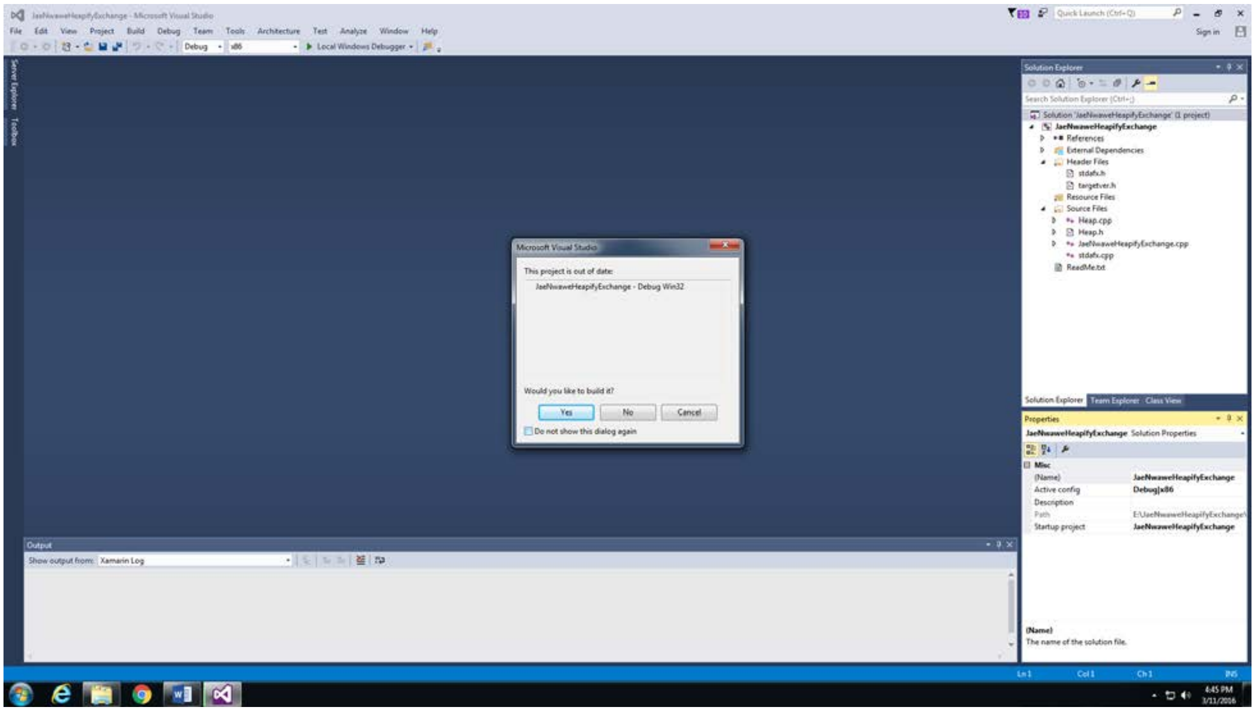

JaeNwaweHeapifyExchange - Microsoft Visual Studio
File Edit View Project Build Debug Team Tools Architecture Test Analyze Window Help
Debug x86 Local Windows Debugger
Output
Show output from: Xamarin Log
Ready
Document1 - Word
FILE HOME INSERT DESIGN PAGE LAYOUT REFERENCES MAILINGS REVIEW VIEW MathType ACROBAT
Document Recovery
Available Files
483 Notes [Original]
Close
PAGE 3 OF 3 128 WORDS
6:45 PM
3/11/2016

Word has recovered the following files. Save the ones you wish to keep.

a. If this toolbar is not visible you can open it using “View” or pressing CTL+ALT+L

6. Dr. Lee, pls press the green triangle that says “Local Windows Debugger”.

6. Visual studio will prompt you to build it since it is on a new system. Select Yes.



Microsoft Visual Studio

This project is out of date:

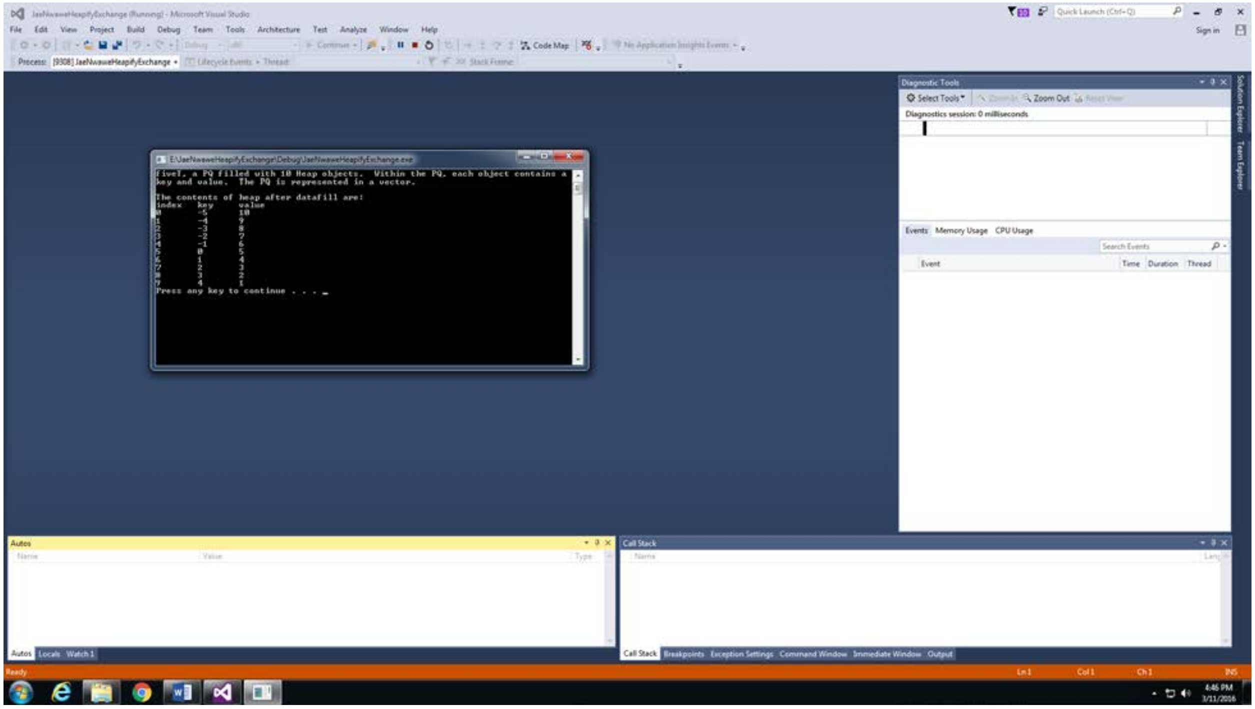
JaeNwaweHeapifyExchange - Debug Win32

Would you like to build it?

Yes | No | Cancel

□ Do not show this dialog again

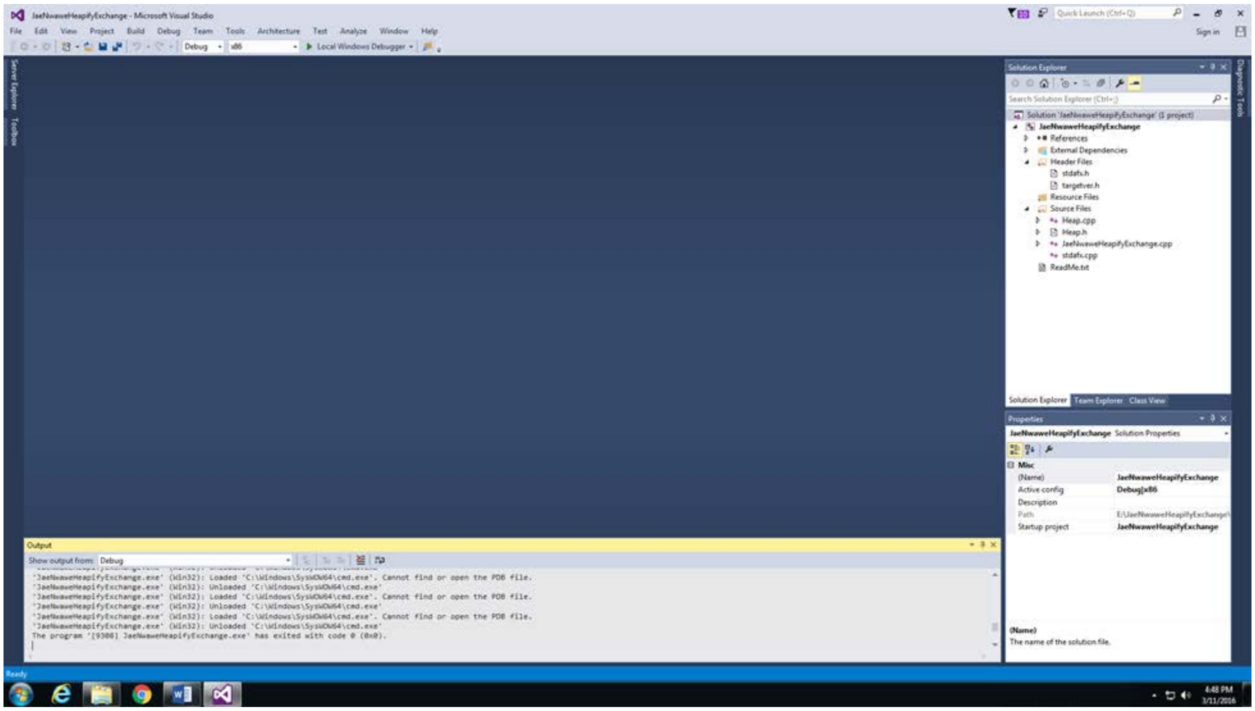
7. The program will start to run.



E:\JaeNwaweHeapifyExchange\Debug\JaeNwaweHeapifyExchange.exe

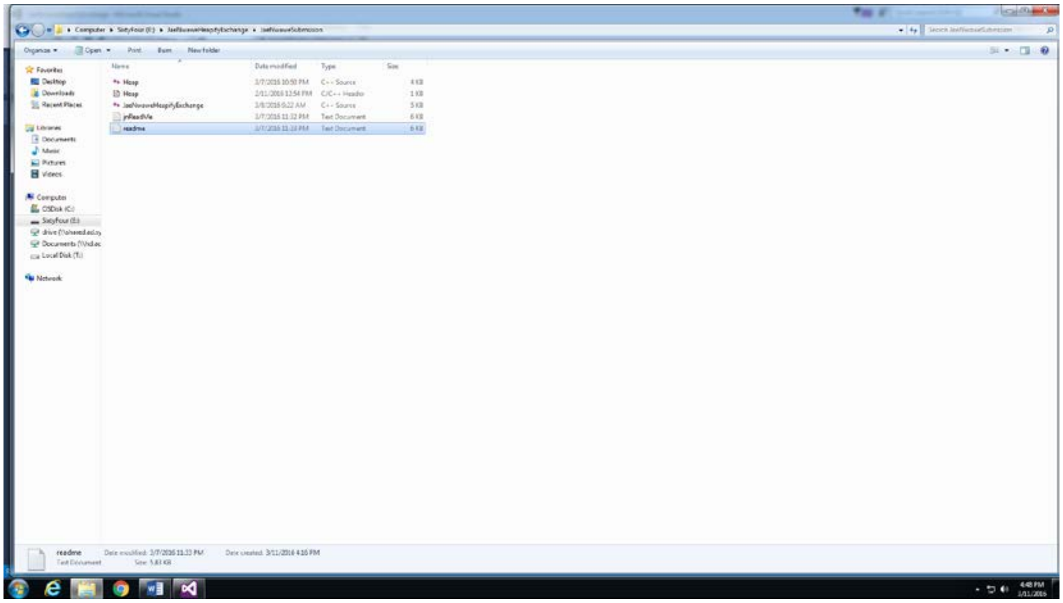
```
five1, a PQ filled with 10 Heap objects.  Within the PQ, each object contains a
key and value.  The PQ is represented in a vector.

The contents of heap after datafill are:
index    key     value
0        -5      10
1        -4      9
2        -3      8
3        -2      7
4        -1      6
5        0       5
6        1       4
7        2       3
8        3       2
9        4       1
Press any key to continue . . . _
```

8. Please, hit any button to see the introduction and all 12 test cases. The program will automatically close after Test case 12 and you hit a button as shown below.



```
'JaeNwaweHeapifyExchange.exe' (Win32): Loaded 'C:\Windows\SysWOW64\cmd.exe'. Cannot find or open the PDB file.
'JaeNwaweHeapifyExchange.exe' (Win32): Unloaded 'C:\Windows\SysWOW64\cmd.exe'
'JaeNwaweHeapifyExchange.exe' (Win32): Loaded 'C:\Windows\SysWOW64\cmd.exe'. Cannot find or open the PDB file.
'JaeNwaweHeapifyExchange.exe' (Win32): Unloaded 'C:\Windows\SysWOW64\cmd.exe'
'JaeNwaweHeapifyExchange.exe' (Win32): Loaded 'C:\Windows\SysWOW64\cmd.exe'. Cannot find or open the PDB file.
'JaeNwaweHeapifyExchange.exe' (Win32): Unloaded 'C:\Windows\SysWOW64\cmd.exe'
The program '[9308] JaeNwaweHeapifyExchange.exe' has exited with code 0 (0x0).
```

9. The readme.txt file is in the JaeNwaweSubmission folder and was also emailed to you.

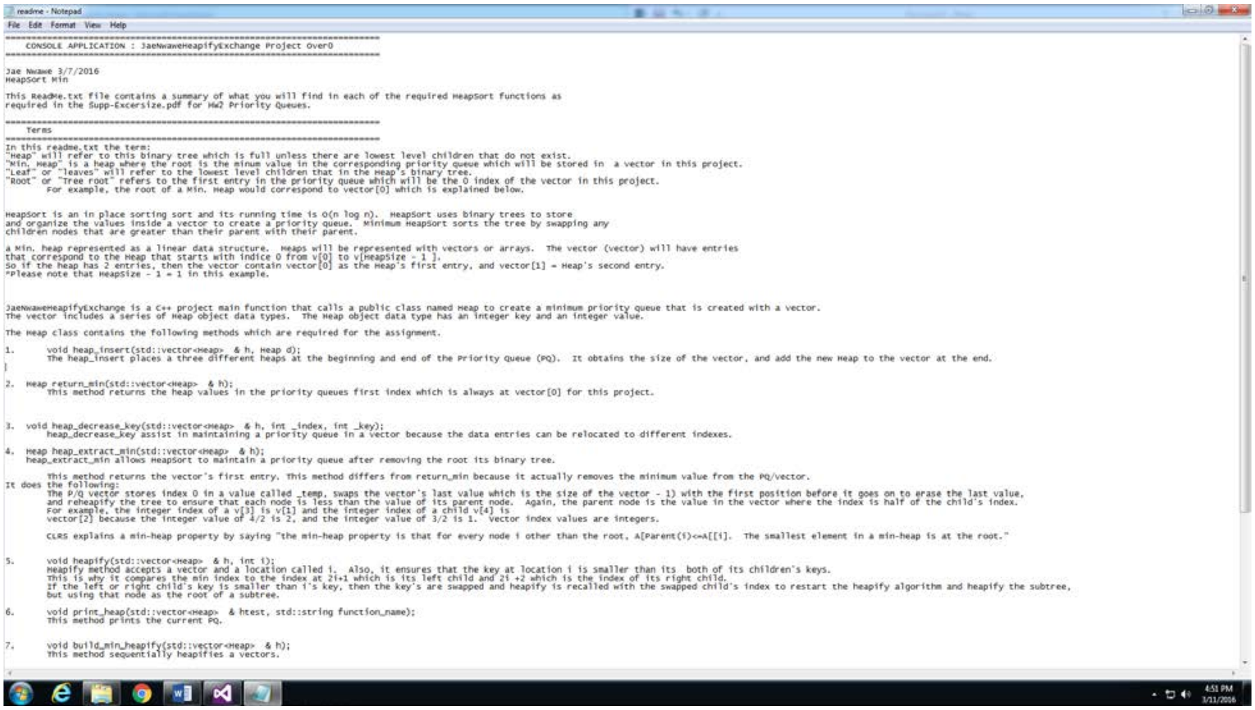


Computer ▸ SixtyFour (E:) ▸ JaeNwaweHeapifyExchange ▸ JaeNwaweSubmission

Name	Date modified	Type	Size
Heap	3/7/2016 10:50 PM	C++ Source	4 KB
Heap	3/11/2016 12:54 PM	C/C++ Header	1 KB
JaeNwaweHeapifyExchange	3/8/2016 5:22 AM	C++ Source	5 KB
jnReadMe	3/7/2016 11:32 PM	Text Document	6 KB
readme	3/7/2016 11:33 PM	Text Document	6 KB

10. “readme.txt” can be opened with notepad or notepad++ as demonstrated below.
    E:\JaeNwaweHeapifyExchange\JaeNwaweSubmssion\ readme.txt .
    (The flash drive that I used was assigned E :). Your drive number may be different.

    *Please, note that the “readme.txt” E:\JaeNwaweHeapifyExchange and dated 3/5/2016 was automatically generated by Visual Studio and does not contain the Heapsort logic that I would like to submit.



```
========================================================================
    CONSOLE APPLICATION : JaeNwaweHeapifyExchange Project Over0
========================================================================

Jae Nwawe 3/7/2016
HeapSort Min

This ReadMe.txt file contains a summary of what you will find in each of the required HeapSort functions as
required in the Supp-Excersize.pdf for HW2 Priority Queues.

========================================================================
    Terms
========================================================================
In this readme.txt the term:
"Heap" will refer to this binary tree which is full unless there are lowest level children that do not exist.
"Min. Heap" is a heap where the root is the minum value in the corresponding priority queue which will be stored in  a vector in this project.
"Leaf" or "leaves" will refer to the lowest level children that in the Heap's binary tree.
"Root" or "Tree root" refers to the first entry in the priority queue which will be the 0 index of the vector in this project.
        For example, the root of a Min. Heap would correspond to vector[0] which is explained below.


HeapSort is an in place sorting sort and its running time is O(n log n).  HeapSort uses binary trees to store
and organize the values inside a vector to create a priority queue.  Minimum HeapSort sorts the tree by swapping any
children nodes that are greater than their parent with their parent.

a Min. heap represented as a linear data structure.  Heaps will be represented with vectors or arrays.  The vector (vector) will have entries
that correspond to the Heap that starts with indice 0 from v[0] to v[HeapSize - 1 ].
So if the heap has 2 entries, then the vector contain vector[0] as the Heap's first entry, and vector[1] = Heap's second entry.
*Please note that HeapSize - 1 = 1 in this example.



JaeNwaweHeapifyExchange is a C++ project main function that calls a public class named Heap to create a minimum priority queue that is created with a vector.
The vector includes a series of Heap object data types.  The Heap object data type has an integer key and an integer value.

The Heap class contains the following methods which are required for the assignment.

1.      void heap_insert(std::vector<Heap>  & h, Heap d);
        The heap_insert places a three different heaps at the beginning and end of the Priority Queue (PQ).  It obtains the size of the vector, and add the new Heap to the vector at the end.


2.  Heap return_min(std::vector<Heap>  & h);
        This method returns the heap values in the priority queues first index which is always at vector[0] for this project.


3.  void heap_decrease_key(std::vector<Heap>  & h, int _index, int _key);
        heap_decrease_key assist in maintaining a priority queue in a vector because the data entries can be relocated to different indexes.

4.  Heap heap_extract_min(std::vector<Heap>  & h);
    heap_extract_min allows HeapSort to maintain a priority queue after removing the root its binary tree.

        This method returns the vector's first entry. This method differs from return_min because it actually removes the minimum value from the PQ/vector.
It does the following:
        The P/Q vector stores index 0 in a value called _temp, swaps the vector's last value which is the size of the vector - 1) with the first position before it goes on to erase the last value,
        and reheapify the tree to ensure that each node is less than the value of its parent node.  Again, the parent node is the value in the vector where the index is half of the child's index.
        For example, the integer index of a v[3] is v[1] and the integer index of a child v[4] is
        vector[2] because the integer value of 4/2 is 2, and the integer value of 3/2 is 1.  Vector index values are integers.

        CLRS explains a min-heap property by saying "the min-heap property is that for every node i other than the root, A[Parent(i)<=A[[i].  The smallest element in a min-heap is at the root."


5.      void heapify(std::vector<Heap>  & h, int i);
        Heapify method accepts a vector and a location called i.  Also, it ensures that the key at location i is smaller than its  both of its children's keys.
        This is why it compares the min index to the index at 2i+1 which is its left child and 2i +2 which is the index of its right child.
        If the left or right child's key is smaller than i's key, then the key's are swapped and heapify is recalled with the swapped child's index to restart the heapify algorithm and heapify the subtree,
        but using that node as the root of a subtree.

6.      void print_heap(std::vector<Heap>  & htest, std::string function_name);
        This method prints the current PQ.


7.      void build_min_heapify(std::vector<Heap>  & h);
        This method sequentially heapifies a vectors.
```