

Dr. Wang

Jeremy Scheuerman

COSC 220

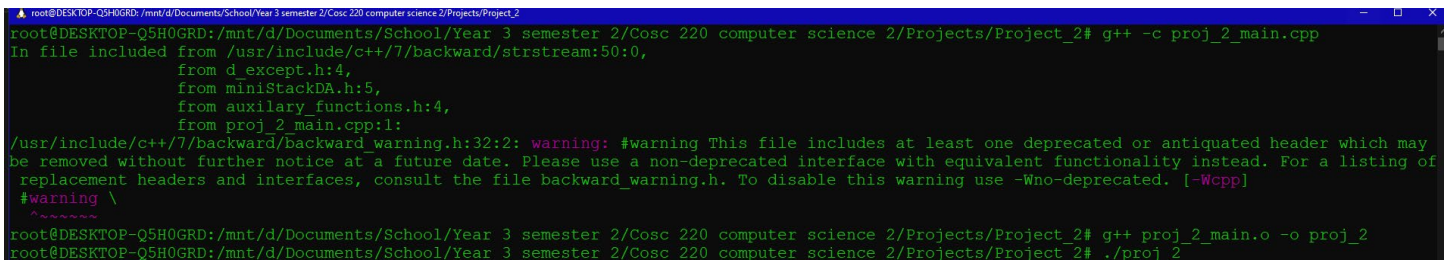
10 April 2021

Project 2 writeup

Testing Plan (all recored in this pdf)

The plan is to make sure that I can add remove print and print top for all 4 implementations of stack and all 4 implementations of queue, I shall try to test it all in one go to make sure I do not get segfaults like I did with the last project. I shall post the output in this pdf.

Building in linux environment



```
root@DESKTOP-Q5H0GRD: /mnt/d/Documents/School/Year 3 semester 2/Cosc 220 computer science 2/Projects/Project_2# g++ -c proj_2_main.cpp
In file included from /usr/include/c++/7/backward/strstream:50:0,
                 from d_except.h:4,
                 from miniStackDA.h:5,
                 from auxiliary_functions.h:4,
                 from proj_2_main.cpp:1:
/usr/include/c++/7/backward/backward_warning.h:32:2: warning: #warning This file includes at least one deprecated or antiquated header which may
be removed without further notice at a future date. Please use a non-deprecated interface with equivalent functionality instead. For a listing of
replacement headers and interfaces, consult the file backward_warning.h. To disable this warning use -Wno-deprecated. [-Wcpp]
#warning \
~~~~~
root@DESKTOP-Q5H0GRD: /mnt/d/Documents/School/Year 3 semester 2/Cosc 220 computer science 2/Projects/Project_2# g++ proj_2_main.o -o proj_2
root@DESKTOP-Q5H0GRD: /mnt/d/Documents/School/Year 3 semester 2/Cosc 220 computer science 2/Projects/Project_2# ./proj_2
```

(warning about header files dosen't affect running of program

Stack Vector

Add elements

```

1
Choose the implementation you would like to use
1: Vector
2: Dynamic Array
3: List
4: Doubly Linked List
1

Choose one of the following actions
1: add element
2: Remove element
3: Print
4: Print top element
5: to exit
1

Please enter a value to add to the stack
23

Choose one of the following actions
1: add element
2: Remove element
3: Print
4: Print top element
5: to exit
1

Please enter a value to add to the stack
54

Choose one of the following actions
1: add element
2: Remove element
3: Print
4: Print top element
5: to exit

```

Print and Print top

```

3
Stack (VT): 54 23

Choose one of the following actions
1: add element
2: Remove element
3: Print
4: Print top element
5: to exit
4

Top (VT):54

Choose one of the following actions
1: add element
2: Remove element
3: Print
4: Print top element
5: to exit

```

Remove elements (and do it until exception handling kicks in

```

Choose one of the following actions
1: add element
2: Remove element
3: Print
4: Print top element
5: to exit
2

```

```

Choose one of the following actions
1: add element
2: Remove element
3: Print
4: Print top element
5: to exit
2

```

```

miniStack pop is empty

```

```

Choose one of the following actions
1: add element
2: Remove element
3: Print
4: Print top element
5: to exit
3

```

```

The stack (VT) is empty

```

```

Choose one of the following actions
1: add element
2: Remove element
3: Print
4: Print top element
5: to exit

```

```

Top (VT):miniStack top is empty

```

```

Choose one of the following actions
1: add element
2: Remove element
3: Print
4: Print top element
5: to exit

```

As we can see exceptions were raised and a message similar showed up when trying to print empty stack (same with printing the top)

Exit and tried again with a different implementation

```

Repeat with a different implementation?
Y for yes and N for no
Y

Enter 1 for Stack or 2 for Queue

```

Try it again for stack dynamic array also see input validation in the menu

```

Enter 1 for Stack or 2 for Queue

1
Choose the implementation you would like to use
1: Vector
2: Dynamic Array
3: List
4: Doubly Linked List
2

Choose one of the following actions
1: add element
2: Remove element
3: Print
4: Print top element
5: to exit
1

Please enter a value to add to the stack
23

Choose one of the following actions
1: add element
2: Remove element
3: Print
4: Print top element
5: to exit
134

Choose one of the following actions
1: add element
2: Remove element
3: Print
4: Print top element
5: to exit

```

Removed Element and saw exception handling messages occur

```

root@DESKTOP-Q5H0GRD: /mnt/d/Documents/School/Year 3 semester 2/Cosc
Stack(DA): 23

Choose one of the following actions
1: add element
2: Remove element
3: Print
4: Print top element
5: to exit
4

Top (DA):23

Choose one of the following actions
1: add element
2: Remove element
3: Print
4: Print top element
5: to exit
2

Choose one of the following actions
1: add element
2: Remove element
3: Print
4: Print top element
5: to exit
2

miniStack top is empty

Choose one of the following actions
1: add element
2: Remove element
3: Print
4: Print top element
5: to exit
4

Top (DA):miniStack top is empty

Choose one of the following actions
1: add element
2: Remove element
3: Print
4: Print top element
5: to exit
3

The stack (DA) is empty

Choose one of the following actions
1: add element
2: Remove element
3: Print

```

Keep going and do it again with STL list

```

5: to exit
5

Repeat with a different implementation?
Y for yes and N for no
Y

Enter 1 for Stack or 2 for Queue
1
Choose the implementation you would like to use
1: Vector
2: Dynamic Array
3: List
4: Doubly Linked List
3

Choose one of the following actions
1: add element
2: Remove element
3: Print
4: Print top element
5: to exit

```

Added 3 elements to the stack and printed it out

```

Please enter a value to add to the stack
2

Choose one of the following actions
1: add element
2: Remove element
3: Print
4: Print top element
5: to exit
1

Please enter a value to add to the stack
76

Choose one of the following actions
1: add element
2: Remove element
3: Print
4: Print top element
5: to exit
1

Please enter a value to add to the stack
56

Choose one of the following actions
1: add element
2: Remove element
3: Print
4: Print top element
5: to exit
3

Stack(LT): 56 76 2

Choose one of the following actions
1: add element
2: Remove element
3: Print
4: Print top element
5: to exit

```

Remove them all and show exception messages happen for deleting and printing

```

root@DESKTOP-Q3H0CKB:/mnt/d/Documents/School/Year 3 semester 2/COSC 220
4: Print top element
5: to exit
2

Choose one of the following actions
1: add element
2: Remove element
3: Print
4: Print top element
5: to exit
2

Choose one of the following actions
1: add element
2: Remove element
3: Print
4: Print top element
5: to exit
2

Choose one of the following actions
1: add element
2: Remove element
3: Print
4: Print top element
5: to exit
2

miniStack pop is empty

Choose one of the following actions
1: add element
2: Remove element
3: Print
4: Print top element
5: to exit
3

The stack (LT) is empty

Choose one of the following actions
1: add element
2: Remove element
3: Print
4: Print top element
5: to exit
4

Top (LT):miniStack top is empty

Choose one of the following actions
1: add element
2: Remove element

```

Repeat process with doubly linked list

```

4: Print top element
5: to exit
5

Repeat with a different implementation?
Y for yes and N for no
Y

Enter 1 for Stack or 2 for Queue
1
Choose the implementation you would like to use
1: Vector
2: Dynamic Array
3: List
4: Doubly Linked List
4

Choose one of the following actions
1: add element
2: Remove element
3: Print
4: Print top element
5: to exit

```

Add element remove and show error messages

```

4

Choose one of the following actions
1: add element
2: Remove element
3: Print
4: Print top element
5: to exit
1

Please enter a value to add to the stack
89

Choose one of the following actions
1: add element
2: Remove element
3: Print
4: Print top element
5: to exit
2

Choose one of the following actions
1: add element
2: Remove element
3: Print
4: Print top element
5: to exit
2

miniStack top is empty

Choose one of the following actions
1: add element
2: Remove element
3: Print
4: Print top element
5: to exit

```

```

3

The stack is empty

Choose one of the following actions
1: add element
2: Remove element
3: Print
4: Print top element
5: to exit

```

Exit now test process with Queues starting with queue vector


```

5
Repeat with a different implementation?
Y for yes and N for no
Y
Enter 1 for Stack or 2 for Queue
2
Choose the implementation you would like to use
1: Vector
2: Dynamic Array
3: List
4: Doubly Linked List
1
Choose one of the following actions
1: add element
2: Remove element
3: Print
4: Print top element
5: to exit

```

Add some elements and print the queue and the queue front

```

Please enter a value to add to the queue
54
Choose one of the following actions
1: add element
2: Remove element
3: Print
4: Print top element
5: to exit
1
Please enter a value to add to the queue
87
Choose one of the following actions
1: add element
2: Remove element
3: Print
4: Print top element
5: to exit
3
Queue (VT): 54 87
Choose one of the following actions
1: add element
2: Remove element
3: Print
4: Print top element
5: to exit
4
Front (VT):54
Choose one of the following actions
1: add element
2: Remove element
3: Print
4: Print top element
5: to exit
1

```

Remove elements and trigger exception for empty stack and print exception messages

```

4
Front (VT):54
Choose one of the following actions
1: add element
2: Remove element
3: Print
4: Print top element
5: to exit
2

Choose one of the following actions
1: add element
2: Remove element
3: Print
4: Print top element
5: to exit
2

Choose one of the following actions
1: add element
2: Remove element
3: Print
4: Print top element
5: to exit
2

miniQueue front is empty

Choose one of the following actions
1: add element
2: Remove element
3: Print
4: Print top element
5: to exit
3

The queue (VT) is empty

Choose one of the following actions
1: add element
2: Remove element
3: Print
4: Print top element
5: to exit
4

Front (VT):miniQueue front is empty

Choose one of the following actions
1: add element
2: Remove element
3: Print

```

Retry with queue dynamic array

```

4: Print top element
5: to exit
5

Repeat with a different implementation?
Y for yes and N for no
Y

Enter 1 for Stack or 2 for Queue
2

Choose the implementation you would like to use
1: Vector
2: Dynamic Array
3: List
4: Doubly Linked List
2

Choose one of the following actions
1: add element
2: Remove element
3: Print
4: Print top element
5: to exit

```

Added some elements and printed them out

```

Select root@DESKTOP-Q5H0GRD: /mnt/d/Documents/School/Year 3 semester 2/Cosc
Please enter a value to add to the queue
1

Choose one of the following actions
1: add element
2: Remove element
3: Print
4: Print top element
5: to exit
1

Please enter a value to add to the queue
45

Choose one of the following actions
1: add element
2: Remove element
3: Print
4: Print top element
5: to exit
3

Queue (DA): 23 1 45

Choose one of the following actions
1: add element
2: Remove element
3: Print
4: Print top element
5: to exit
4

Front (DA):23

Choose one of the following actions
1: add element
2: Remove element
3: Print
4: Print top element
5: to exit

```

Remove elements until nothing is left and exception messages appear

```

Select root@DESKTOP-Q5H0GRD: /mnt/d/Documents/School/Year 3 seme
3: Print
4: Print top element
5: to exit
2

Choose one of the following actions
1: add element
2: Remove element
3: Print
4: Print top element
5: to exit
2

miniQueue dequeue is empty

Choose one of the following actions
1: add element
2: Remove element
3: Print
4: Print top element
5: to exit
3

The queue (DA) is empty

Choose one of the following actions
1: add element
2: Remove element
3: Print
4: Print top element
5: to exit
4

Front (DA):miniQueue front is empty

Choose one of the following actions
1: add element
2: Remove element
3: Print
4: Print top element
5: to exit

```

Restart with stl list

```

3: Print
4: Print top element
5: to exit
5

Repeat with a different implementation?
Y for yes and N for no
Y

Enter 1 for Stack or 2 for Queue
2

Choose the implementation you would like to use
1: Vector
2: Dynamic Array
3: List
4: Doubly Linked List
2

Choose one of the following actions
1: add element
2: Remove element
3: Print
4: Print top element
5: to exit
1

Please enter a value to add to the queue
23

Choose one of the following actions

```

```

5: to exit
1

Please enter a value to add to
76

Choose one of the following actions
1: add element
2: Remove element
3: Print
4: Print top element
5: to exit
1

Please enter a value to add to
40

Choose one of the following actions
1: add element
2: Remove element
3: Print
4: Print top element
5: to exit
3

Queue(LT): 23 76 40

Choose one of the following actions
1: add element
2: Remove element
3: Print
4: Print top element
5: to exit
4

Front (LT):23

Choose one of the following actions
1: add element
2: Remove element
3: Print
4: Print top element

```

Finally test with queue doubly linked list

```

5
Repeat with a different implementation?
Y for yes and N for no
Y
Enter 1 for Stack or 2 for Queue
2
Choose the implementation you would like to use
1: Vector
2: Dynamic Array
3: List
4: Doubly Linked List
4

Choose one of the following actions
1: add element
2: Remove element
3: Print
4: Print top element
5: to exit
|

```

Add some elements to it

```
root@DESKTOP-Q5H0GRD: /mnt/d/Documents/School/Year 3 semester 2/Cosc 220
2: Remove element
3: Print
4: Print top element
5: to exit
1

Please enter a value to add to the queue
34

Choose one of the following actions
1: add element
2: Remove element
3: Print
4: Print top element
5: to exit
1

Please enter a value to add to the queue
76

Choose one of the following actions
1: add element
2: Remove element
3: Print
4: Print top element
5: to exit
1

Please enter a value to add to the queue
4

Choose one of the following actions
1: add element
2: Remove element
3: Print
4: Print top element
5: to exit
3

Queue(DL): 34 76 4

Choose one of the following actions
1: add element
2: Remove element
3: Print
4: Print top element
5: to exit
4

Front (DL):34

Choose one of the following actions
1: add element
2: Remove element
3: Print
```

Now remove them all and make sure remove print and print front exceptions work

```

4: Print top element
5: to exit
2

Choose one of the following actions
1: add element
2: Remove element
3: Print
4: Print top element
5: to exit
2

miniQueue top is empty

Choose one of the following actions
1: add element
2: Remove element
3: Print
4: Print top element
5: to exit
3

The queue (DL) is empty

Choose one of the following actions
1: add element
2: Remove element
3: Print
4: Print top element
5: to exit
4

Front (DL):miniQueue top is empty

Choose one of the following actions
1: add element
2: Remove element
3: Print
4: Print top element
5: to exit

```

Exit and don't repeat

```

Repeat with a different implementation?
Y for yes and N for no
n

root@DESKTOP-Q5H0GRD:/mnt/d/Documents/School/Year 3 semester 2/Cosc 220 computer science 2/Projects/Project_2# |

```

All possible cases tested no errors occurred!

