

# Lab 8



## Required tools: Arduino / Arduino IDE

### [Step 1: Creating The Stack Class]

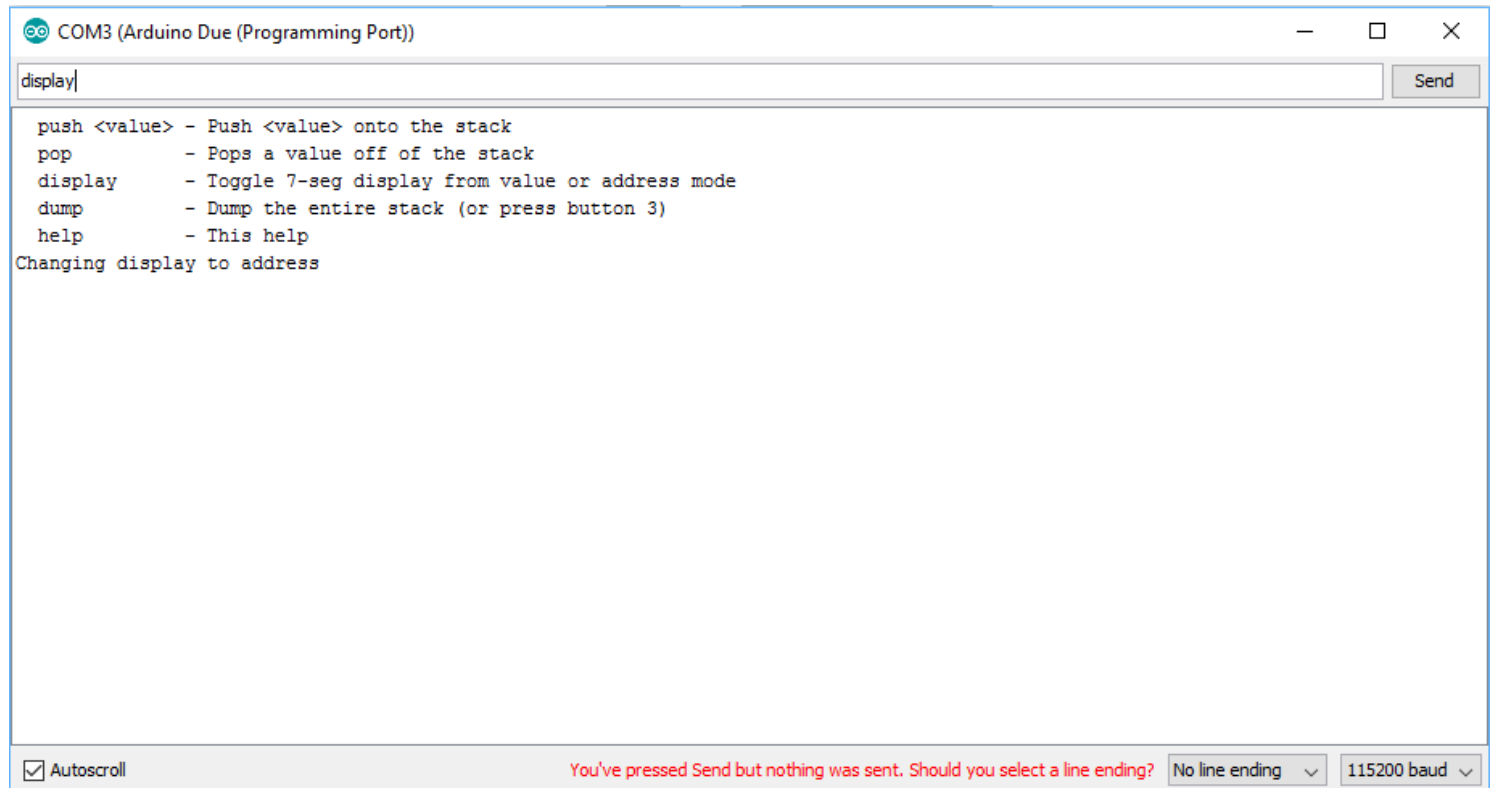
1. Download [lab8stud.ino](#)
2. Create a class called **Stack** with the following members:
  - a. A private integer array called **stack** that contains **STACK\_SIZE** elements. You will only use this array in your constructor and st functions. Do NOT use it directly with any of your other member functions!
  - b. A private integer pointer called **stack\_pointer**.
  - c. A public constructor. This constructor will set the **stack\_pointer** to the *bottom* of the stack. Remember that stacks grow from bottom to top!
  - d. A public void function called **push**. This function takes one integer parameter which is the value to be pushed on the stack. This function will push the given value onto the stack.
  - e. A public function called **pop**. This function takes no parameters, but returns an integer. This function will pop the value given by the stack pointer.
  - f. A public function called **sp**. This function takes no parameters, but returns an integer pointer. This function will return the stack pointer variable.
  - g. A public function called **st**. This function takes no parameters, but returns an integer pointer. This function will return the TOP of the stack array. Remember that the stack grows from bottom to top. This function always returns the address of the top (the first element of the stack array).

**NOTE:** Notice that there are no if statements, conditionals, or any boolean logic for a stack. It is up to the user to make sure that they do not exceed the bounds of the stack. As such, you must not have any sanity checks in your code!

### [Step 2: Testing Your Stack]

1. Since you're dealing with memory locations, **it is very likely that you will crash your Arduino**. If you only get one digit on the 7-segment display, and the serial monitor stops responding, your Arduino probably crashed. You may re-upload your sketch to reset it.
2. Upload your sketch to the Arduino.
3. Open the Serial Monitor.
4. You will notice that the 7-segment display gives a value.

- a. You should see ---- (four dashes) if the stack pointer is correctly pointing to the bottom of the stack.
- b. It has two different modes that you can toggle in the serial monitor with the *display* command. The default mode gives you the value at the current stack pointer. The address mode gives you the last four hex digits of the address the stack pointer points to.



```
display| Send
```

```
push <value> - Push <value> onto the stack
pop          - Pops a value off of the stack
display     - Toggle 7-seg display from value or address mode
dump        - Dump the entire stack (or press button 3)
help        - This help
Changing display to address
```

Autoscroll      You've pressed Send but nothing was sent. Should you select a line ending?      No line ending      115200 baud

5. Button A1 (leftmost button) pushes a random value onto your stack. Button A2 (middle button) pops a value off of the stack. (While I added logic to avoid crashing your Arduino, you should NEVER pop before you push). Button A3 (rightmost button) will dump your stack on the serial monitor. You can also use the serial monitor commands "push <value>" and "pop" to push and pop your stack.
6. Make sure that your stack pointer moves as you push and pop. You can see what the stack pointer points to by the asterisk in the "dump" on the serial monitor.

```
COM3 (Arduino Due (Programming Port))
dump]
Pushed 14, pointer is at 0x2007096c
Pushed 15, pointer is at 0x20070968
Pushed 17, pointer is at 0x20070964
Dumping stack (asterisk = stack pointer):

Address      Value
0x20070934   0
0x20070938   0
0x2007093c   0
0x20070940   0
0x20070944   0
0x20070948   0
0x2007094c   0
0x20070950   0
0x20070954   0
0x20070958   0
0x2007095c   0
0x20070960   0
*0x20070964  17
0x20070968  15
0x2007096c  14
0x20070970  12

 Autoscroll
No line ending
115200 baud
```

7. If your stack pointer moves appropriately and all of the values are given/taken appropriately, then you're finished!

[You are finished with this lab!]

Submit your .ino file as a .txt file.

Points 100

Submitting a file upload

File Types txt

Due	For	Available from	Until
Oct 17, 2017	Everyone	Oct 13, 2017 at 7:58am	Oct 17, 2017 at 11:59pm

**Lab 8**

Criteria	Ratings						Pts
Stack pointer is correct at construction	<b>10.0 pts Full Marks</b>			<b>0.0 pts No Marks</b>			10.0 pts
push() function -Make sure there are no references to the stack array	<b>35.0 pts Full Marks</b>	<b>24.0 pts Minor problems</b>	<b>18.0 pts Some Problems</b>	<b>12.0 pts More Problems</b>	<b>5.0 pts Major problems</b>	<b>0.0 pts No Marks</b>	35.0 pts
pop() function -Make sure there are no references to the stack array	<b>35.0 pts Full Marks</b>	<b>24.0 pts Minor problems</b>	<b>18.0 pts Some Problems</b>	<b>12.0 pts More Problems</b>	<b>5.0 pts Major problems</b>	<b>0.0 pts No Marks</b>	35.0 pts
sp() function	<b>10.0 pts Full Marks</b>			<b>0.0 pts No Marks</b>			10.0 pts
st() function	<b>10.0 pts Full Marks</b>			<b>0.0 pts No Marks</b>			10.0 pts
<b>Total Points: 100.0</b>							

