CS 161 Recitation

Worksheet: Week 9
Weekly quiz will be on Canvas.

1. Write C++ code that creates a jagged 2-dimensional array of variables with type: **double**. The array should be named **jagged** and will have 3 rows. The first and last row will contain 6 **doubles** while the middle row should contain 14 **doubles**. Do not worry about initializing the values.

2. Now write code to deallocate the array that you created in problem 1.

3. Explain why the order of deallocation is important in problem 2.

4. From an implementation perspective, what change do you have to make to the traditional **main** function in order to accept command line arguments? What are the types of the two variables (traditionally named argc and argv) that are passed from the command line?

5. Write code to implement the `generate_sum()` function shown below. The function should compute the sum of all numbers between (and including) **lower_limit** and **upper_limit**. However, the sum should not include numbers that are divisible by 11. For example, if you user input 7 and 15, your code would compute: 7+8+9+10+12+13+14+15 = 88
   You are free to use whichever type of loop that you prefer, but your code needs to use the **continue** statement.

```
int generate_sum() {
   unsigned int lower_limit;
   unsigned int upper_limit;
   unsigned int sum = 0;

   cin >> lower_limit;
   cin >> upper_limit;









   return sum;
}
```

6. A) Describe at least two reasons why students should write their code in a modular format.
   B) Consider the previous homework projects for this class. Has modular design saved you time? If not, consider how you could make use of modularity to save time in Homework #6.