In Class work for Lecture 3/22/2017

Objective: Learn about program optimization

1. Read the documentation for gcc on wcpkneel
   1.1. How many optimization flags are turned on for -O1?
   1.2. How many optimization flags are turned on for -O2?
   1.3. How many optimization flags are turned on for -O3?
   1.4. Are there any other ways to optimize?
   1.5. Pick an optimization flag.
      1.5.1. How would you invoke it individually apart from -Ox?
      1.5.2. What does it do?
2. Download optimize_base.c
   2.1. Compile it with each level of optimization separately
   2.2. Complete the following table:

Optimization Level	How big is the executable file?	How long did it take to run the program?	How many assembly commands are in set_two_dimensions?	How many assembly commands are in get_two_dimension?	How many assembly commands are in main?	Total of set+get+main?	How many calls to set or get are made from the assembly code of main?
-O0							
-O1							
-O2							
-O3							

3.
   3.1. What do these numbers teach you about what optimization does and doesn't do?

(Rock Star of Computer Organization Follow-up on the back)

Rock Star of Computer Organization

4.

4.1.Manually Inline the set_two_dimension and get_two_dimension methods manually in main.
4.2.Repeat the experiments on the above table.

Optimization Level	How big is the executable file?	How long did it take to run the program?	How many assembly commands are in set_two_dimensions?	How many assembly commands are in get_two_dimension?	How many assembly commands are in main?	Total of set+get+main?	How many calls to set or get are made from the assembly code of main?
-O0			N/A	N/A		N/A	N/A
-O1			N/A	N/A		N/A	N/A
-O2			N/A	N/A		N/A	N/A
-O3			N/A	N/A		N/A	N/A

4.3.How did the results compare?
4.4.Were you able to improve performance by inlining this function manually?