Fall 2015
CS3413
Lab4

Memory Allocation

Write a small program to allocate memory dynamically using *malloc()* in a potential infinite while loop. Write your program such that:

1. Each call to malloc allocates 1K of memory.
2. You can control how many iterations of the loop you perform.
3. Do NOT free memory that you allocate ☺

free is a command that displays used and available memory in your system. Read the man page for more information. Explore the effect of your program on the available memory using the ***free*** command. You will want to insert some sleep() calls so you can see your results when using ***free***.

For each of the questions below you should submit your program (ie. q1.c, q2.c, q3.c, and q4.c) to D2L where you include your written answers to any questions as a comment in the C program.

Questions

1) Run the command ***free –o*** before running your memory allocation program, during and after. Explain the behavior with respect to memory utilization.

2) Modify your program so that you can record the difference in addresses between calls to malloc. Set your loop to 100000 iterations. Check the difference between calls to malloc and get the average. What do you notice about the result? What does this tell you about memory allocation?

3) Modify your program from question 1) so that you *free()* some of the memory you allocate. Does any of this memory get "reallocated" (ie. you see this address again returned from *malloc()*)?

4) Let's be malicious! Can you modify your program so that you get an Out Of Memory exception? It is harder than it sounds! What did you have to do in order to get your program to run out of memory? What did you observe about the system's memory?