PART II: WRITTEN ASSIGNMENTS

1) $2n^2 + 5 \ is \ O(n^2)$

   $Let \ c = 7, \ m = 1$

   $Notice \ that \ 2 * n^2 \ \leq 2 * n^2 \ and \ 5 \ \leq 5 * n^2 \ when \ n \ \geq 1$

   $Therefore \ 2 * n^2 + 7 \leq (2 + 5) * n^2 \ when \ n \geq 1$

   $Thus, \ we \ show \ that \ 2n^2 + 5 \ \leq c * n^2 \ for \ all \ values \ n \geq m$

2) $Proof \ that \ n^4 - 3n^3 \ + 8n^2 \ is \ O(n).$

   $Pick \ c \ = 6 \ and \ m \ = \ 1.$

   $For \ all \ of \ n \ greater \ than \ or \ equal \ to \ one, \ each \ individual \ element \ can \ be \ proven \ to \ be \ less \ than \ or \ equ$

   $Notice \ that \ n^4 \leq 1 * (n^4), \ -3n^3 \leq \ -3 * (n^3), \ and \ 8n^2 \leq 8 * (n^2) \ when \ n \geq 1.$

   $Therefore, it \ can \ be \ said \ that:$

   $n^4 \ + \ -3n^3 \ + 8n^2 \ \leq (1 - 3 + 8) \ * n \ = \ 6 * n \ when \ n \ \geq \ 1.$

   $Furthermore, \ the \ sum \ of \ all \ coefficients \ of \ all \ elements \ is \ equal \ to \ the \ C \ of \ the \ overall \ expression:$

   $1 \ + \ -3 \ + \ 8 \ = \ 6$

   $Hence, \ we \ showed \ that \ n^4 - 3n^3 \ + \ 8n^2 \ \leq \ c * n \ when \ n \ \geq$ 1.

PART III MYSTERY FUNCTIONS

Function A $O(n)$:

Function A has a linear runtime because the single For-Loop iterates through n/2 times, so it grows at a linear rate.

Function B $O(n^2)$:

FnB has a quadratic runtime because the function is a nested For-Loop that will iterate n times in the outer and inner loop, being $n * n$ or n^2 times.

Function C $O(n)$:

FnC has a linear runtime because, although there are nested For-Loops, the inner For-Loop will only run 4 times for any value of n. As the outer For-Loop only runs n times, the run time is $4 * n$, being a linear function.

Function D $O(n^3)$:

FnD has a cubic runtime because the For-Loop will iterate n*n*n times, therefore going through the loop n^3 times.

Function E $O(n \ * \ logn)$:

FnE has an n*log(n) runtime because the outer For-Loop will iterate n times, while the inner While-Loop will iterate 2^x times until the expression becomes greater than n, which would make that while loop log(n). Therefore, the total runtime of fnE would be $O(n * log(n))$.

Function F $O(n^4)$

FnF has a runtime of n^4 because the outer For-Loop will iterate $n * n$ or n^2 times and the inner For-Loop will do the same, so the total runtime will be $n^2 * n^2 = n^4$.

USING FUNCTION_TIMER TO INSPECT THE FUNCTIONS

F1 = Function C

F1 is the second slowest growing function, when compared to the other functions, and grows slightly faster than f6, leading us to believe f1 correlates to fnC. FnC has a runtime of O(n) at the strictest level, but while fnC and fnA both have linear runtimes, fnC runs through $4 * n$ iterations and fnA runs through $n/2$ iterations, causing fnC to have the second slowest runtime matching with f1.

F2 = Function F

F2 is the fastest growing function out of all the functions. When compared to the other functions, f2 grows much faster than any of the other functions, indicating it would be fnF which has a runtime of $O(n^4)$, being the fastest growing function.

F3 = Function B

F3 is the 3rd fastest growing function, which grows substantially faster than f6, f1, and f5, but slower than f4 and f2. With this in mind, f3 matches to fnB, since fnB has a runtime of $O(n^2)$, which is faster than three of the functions, but slower than fnD and fnF, which have runtime functions that are raised to higher powers.

F4 = Function D

F4 is the second fastest growing function, growing slightly slower than f2. This led us to believe f4 is fnD, since fnD has a runtime of $O(n^3)$, which is the second fastest growing function coming in after fnC.

F5 = Function E

F5 is the 3rd slowest growing function, growing faster than f1, and slightly faster than f6, but not faster than the others. Being the 3rd slowest growing, this matches with fnE, since that is also the 3rd slowest growing, as it has a runtime of $O(n * log(n))$ and grows faster than the two linear functions, but slower than the functions that are raised to a power.

<u>F6 = Function A</u>

F6 is the slowest growing function compared to all of the other functions, so it would match up with fnA, which is a linear function. FnA is also the slowest growing function out of all the other functions and has a run time of O(n).