Name: Himanshu Rana Date: 9/18/18

Pledge: “I pledge my honor that I have abided by the Stevens Honor System” – Himanshu Rana (hrana2)

Give the complexity of the following functions. Choose the most appropriate notation from among , , and .

1. **void** **function1**(**int** n) {

**for** (**int** i = 1; i <= n; i++) {

**for** (**int** j = i; j <= n; j += 2) {

cout << "\*";

}

}

}

Answer: (n^2)

1. **void** **function2**(**int** n) {

**int** count = 0;

**for** (**int** i = 1; i \* i <= n; i++) {

count++;

}

cout << count;

}

Answer: ()

1. **void** **function3**(**int** n) {

**int** count = 0;

**for** (**int** i = n/2; i <= n; i++) {

**for** (**int** j = 1; j + n/2 <= n; j++) {

**for** (**int** k = 1; k <= n; k \*= 2) {

count++;

}

}

}

cout << count;

}

Answer: (n^2(log(n)))

1. **void** **function4**(**int** n) {

**int** count = 0;

**for** (**int** i = n/2; i <= n; i++) {

**for** (**int** j = 1; j <= n; j \*= 2) {

**for** (**int** k = 1; k <= n; k \*= 2) {

count++;

}

}

}

cout << count;

}

Answer: (n(log(n))(log(n)))

1. **void** **function5**(**int** n) {

**if** (n % 2 == 0) {

**return**;

}

**for** (**int** i = 1; i <= n; i++) {

**for** (**int** j = 1; j <= n; j++) {

cout << "\*";

**break**;

}

}

}

Answer: (n)

1. **void** **function6**(**int** n) {

**int** count = 0;

**for** (**int** i = 1; i <= n/2; i++) {

**for** (**int** j = 1; j <= n/3; j++) {

**for** (**int** k = 1; k <= n/4; k++) {

count++;

}

}

}

cout << count;

}

Answer: (n^3)

1. **void** **function7**(**int** n) {

**for** (**int** i = 1; i <= n; i++) {

**for** (**int** j = 1; j <= n; j += i) {

cout << "\*";

}

}

}

Answer: (nlogn)

1. **void** **function8**(**int** n) {

**int** i = 1, s = 1;

**while** (s <= n) {

i++;

s += i;

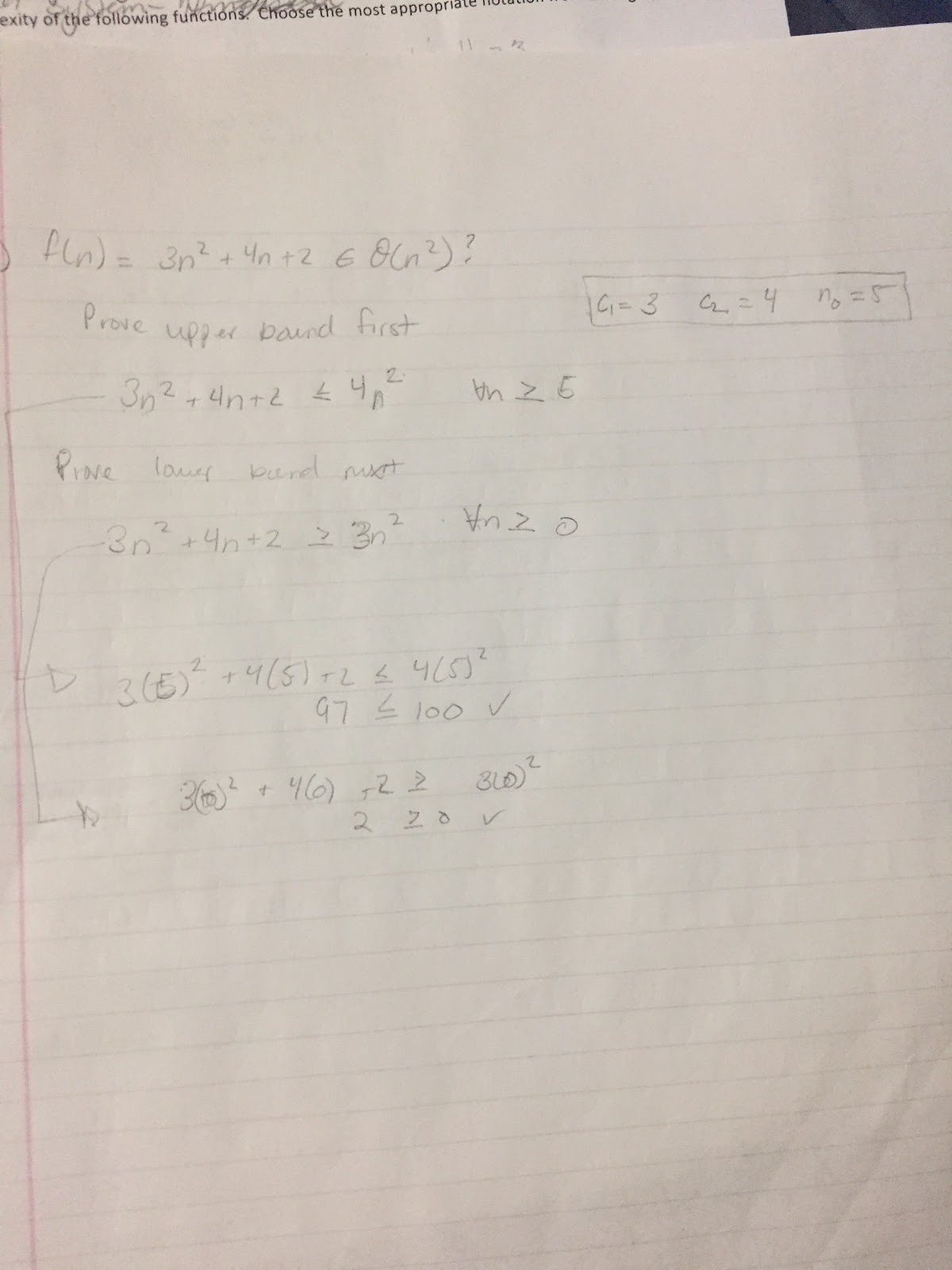
cout << "\*";

}

}

Answer: ()

1. Processing Arrays
   1. Suppose you have an unsorted array of integers of length and want to sum all the elements inside it. What is the running time of your algorithm? → **(n)**
      * To get the sum of all the integers you have to go through the entire array
   2. Suppose you have an unsorted array of integers of length and want to determine if all the values inside are positive. What is the running time of your algorithm? → **(n)**
      * The best case scenario is that the first element in the array is negative and get out of the algorithm
   3. Suppose you have a sorted array of integers of length and want to determine the median value. What is the running time of your algorithm? 🡪 **(1)**
      * Because the array is already sorted all the algorithm has to do is divide n by 2 and add the two numbers up and then divide that by 2 (if n/2 is odd)
2. **T🡪** T / F

If true, prove it by giving *integral* values for the required constants , , and . Choose the tightest values possible for the and constants. If false, show the contradiction.

= 3

= 4

= 5