**Data Structures and Algorithms - Big-Oh Notation**

**Task 1:**

**Specify which function is in big-oh of the other (each question is a pair, so you need to compare the two) and explain why:**

1. *f(n) = n2, g(n) = n + 10*
2. *f(n) = 4n, g(n) = n2*
3. *f(n) = n2, g(n) = 2n*
4. *f(n) = nlog(n), g(n) = log(n)*
5. *f(n) = √n, g(n) = nlog(n)*

**Task 2:**

**Specify, using big-oh notation, the running times of the following functions:**

**1.**

**x = 5 + (15\*20);**

**2.**

**x = 5 + (15\*20);**

**y = 10 – 2;**

**print x - y;**

**3.**

**for x in range (0, n);**

**print x;**

**4.**

**y = 10 – 2;**

**for x in range (0, n);**

**print x;**

**5.**

**void function(int input) {**

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

**cout << “Step” << endl;**

**}**

**}**

**6.**

**void function(int input) {**

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

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

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

**cout << “Step” << endl;**

**}**

**}**

**}**

**}**

**7.**

**void function(int input) {**

**cout << “Step” << endl;**

**}**

**8.**

**void function(int input) {**

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

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

**cout << “Step” << endl;**

**}**

**}**

**}**

**Task 3:**

**From Task 2 on a single graph, rank the functions in order of their big-oh inclusion.**