Name:

Score: ***/ 60***

CS 6212 – Final

(60 points) (120 minutes)

Q 1 – 6 (5 points each): ***[Only write answers, NO explanations]***

Questions 1, 2, 3: What is the time complexity of these algorithms/functions, in terms of n? ***[Sum += y is a short form notation for Sum = Sum + y.]***

```
for (int i = 1 to n) {
  for (int j = i to n) {
    for (int k = j*j to n) {
      Sum += a[i]*b[j]*c[k]
    }
  }
}
```

```
int j = 5
while (j < log n) {
  int k = 5
  while (k < n) {
    Sum += a[j]*b[k]
    k = k^1.3
  }
  j = 1.3 * j
}
```

```
function T(int n)
{
  int n1 = T(n/2)
  int n2 = n1*n1 + n1
  int sum = 0
  for i = 1 to n
    for j = 1 to n
      sum+=n1*i + n2*j
  return sum
}
// Assume T(1) = 1
```

Q4: Give an example of a graph that has all of the following properties. (Give a single graph as the answer.)

(i) It does not have any articulation point.

(ii) It does not have a Hamiltonian cycle

(iii) It does not have a valid vertex coloring with only 2 colors.

Q5: Solve recurrence relation: T(n) = T(n/2) + O(log n).

Q6: Solve recurrence relation: T(n) = 3T(n/2) + O(n log n).

Q7 (10 points): **("Maximum Value But Limited Neighbors")** You are given an array *a[1..n]* of positive numbers and an integer *k*. You have to produce an array *b[1..n]*, such that: (i) For each *j*, *b[j]* is *0* or *1*, (ii) Array *b* has adjacent *1*s at most *k* times, and (iii) $\sum_{j=1}^{n}(a[j] * b[j])$ is maximized. For example, given an array *[100, 300, 400, 50]* and integer *k = 1*, the array *b* can be: *[0 1 1 0]*, which maximizes the sum to be *700*. Or, given an array *[10, 100, 300, 400, 50, 4500, 200, 30, 90]* and *k = 2*, the array *b* can be *[1, 0, 1, 1, 0, 1, 1, 0, 1]* which maximizes the sum to *5500*.

Q8 (10 points): Describe a Branch & Bound algorithm to solve the employee to project assignment reward problem. You are given n employees and n projects. You are also given an array A[i,j] which contains the revenue realized by that assignment of employee i to project j. Describe your B&B solution to maximize the reward.

Q9 (10 points): Prove that the following problem is NP-complete: *Given a graph G, and an integer k, find whether or not graph G has a spanning degree where the maximum degree of any node is k.*

(Hint: Show a reduction from one of the following known NP-complete problems: Vertex Cover, Ham Path or SAT.)