CS 325 Final Exam practice Solution:

1. What is the big O notation of f(n) ?
   F(n) = 2n(n-1)

   O(n^2)

2. What is the time complexity of below pseudocode?

```
IsUnique(A[0..n − 1]):
   for i =0 to n − 2
      for j = i + 1 to n − 1
         if A[i]= A[j]
            return false
   return true
```

   Theta(n^2)

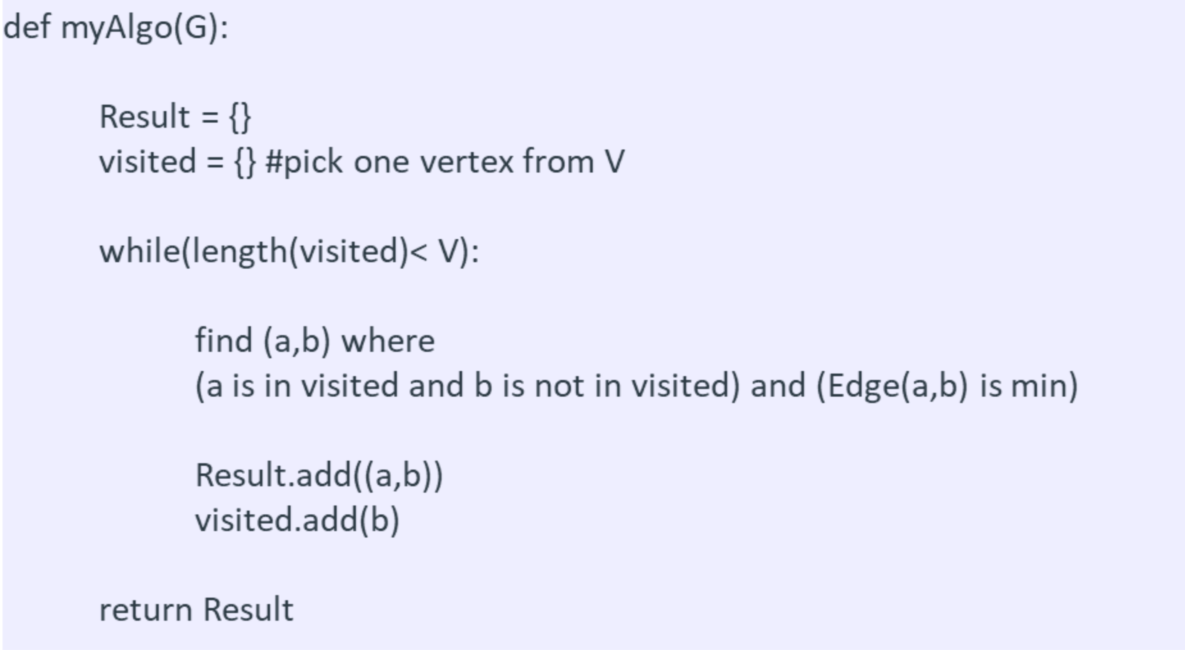
3. What are the definitions of Big O, Big Theta and Big Omega?
   Refer to exploration

4. Given Problem A, that can be verified in polynomial time. If Hamiltonian Cycle problem reduces to Problem A in polynomial time, we can say A is in NP-Complete. (true/false)
   True

5. Every problem in NP can be reduced to NP hard problems? (true/false)
   True

6. How do we show a problem to be NP-Complete?
   Refer to exploration

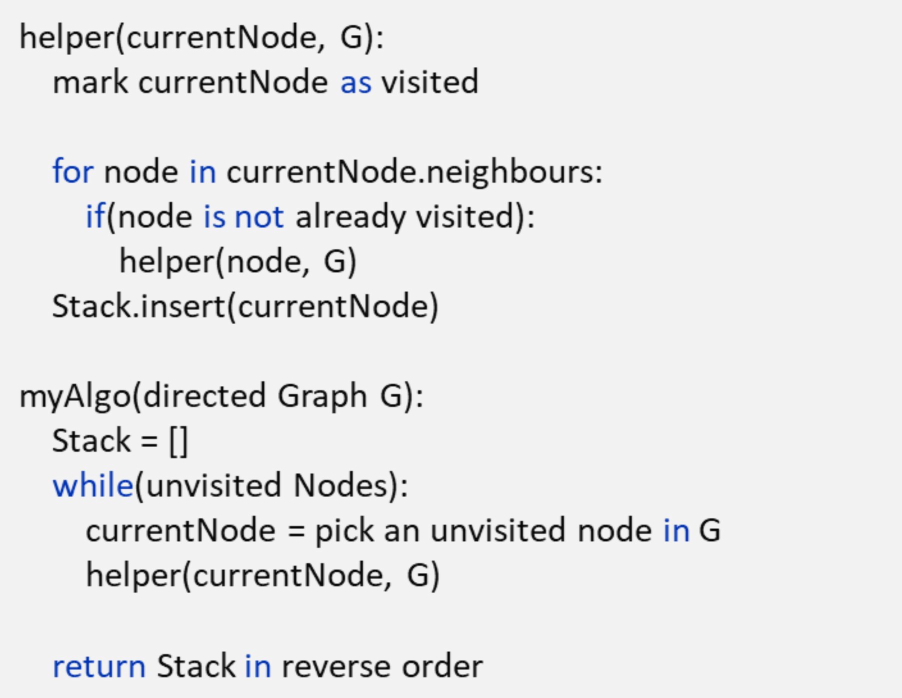
7. The following pseudocode is for which of the algorithms?



```
def myAlgo(G):

    Result = {}
    visited = {} #pick one vertex from V

    while(length(visited)< V):

        find (a,b) where
        (a is in visited and b is not in visited) and (Edge(a,b) is min)

        Result.add((a,b))
        visited.add(b)

    return Result
```

A. Dijkstra
B. Prim’s
C. Topological Sort
D. Kruskal’s

8. The following pseudocode is for which of the algorithms?



```
helper(currentNode, G):
  mark currentNode as visited

  for node in currentNode.neighbours:
    if(node is not already visited):
      helper(node, G)
  Stack.insert(currentNode)

myAlgo(directed Graph G):
  Stack = []
  while(unvisited Nodes):
    currentNode = pick an unvisited node in G
    helper(currentNode, G)

  return Stack in reverse order
```

Which algorithm is myAlgo?

A. Dijkstra

B. Prim's

C. Topological Sort

D. Kruskal's

9. Activity selection problem is given as : You are given a list of activities {a_1, a_2,...a_n} with their start times [s_1, s_2,...s_n] and end times [e_1,e_2,....e_n]. Your goal is to maximize the number of activities that you can perform. You cannot choose overlapping activities. Describe a greedy approach to solve this problem.

   Refer to the exploration

10. Which of the following data structures can be used to implement the Dijkstra algorithm most efficiently?
    Min priority queue
    Max priority queue
    Stack
    Circular queue

11. There are two problems, X and Y. 3SAT reduces to problem X and problem X reduces to Problem Y. (These reductions are poly-time reductions.)
    Is it true that: Y is not in NP-hard.
    False