SSN College of Engineering, Kalavakkam – 603 110
(An Autonomous Institution, Affiliated to Anna University, Chennai)
Department of Computer Science and Engineering
Continuous Assessment Test – III

Degree & Branch:	**B.E. Computer Science and Engineering**		Semester: **4**
Subject Code & Name:	**UCS1403 DESIGN AND ANALYSIS OF ALGORITHMS**		
Academic Year:	**2020-2021**	Batch: 2019-2023	Date: **24-04-2021**
Time: 10:00 am-11:55 am	Answer All Questions		Maximum: 50 Marks

Part A (30 min, $10 \times 2 = 20$ marks)

1. Suppose we have 4-, 3-, and 1-rupee coins. We have to give change for 6 rupees. If we greedily choose the largest coins first, what is the set of coins? What is the optimal set of coins? CO3,K2

   A. $\{4, 1, 1\}, \{3, 3\}$

   B. $\{3, 1, 1, 1\}, \{3, 3\}$

   C. $\{4, 2\}, \{4, 3\}$

   D. $\{4, 1, 1\}, \{3, 3\}$

2. The weights, values, and value per unit weight of 5 items are given below. CO3,K2

w	10	20	30	40	50
v	20	30	66	40	60
v/w	2.0	1.5	2.2	1.0	1.2

   In a knapsack which can carry a weight of at most 100 units, if you are permitted to take fractions of items, what is the maximum value you can pack?

   A. 164

   B. 146

   C. 156

   D. 166

3. In the Parliament of Peaceland, each member has at most three enemies. The parliament needs to be separated into two houses, so that each member has at most one enemy in his own house. An Algorithmist suggested that initially the members be separated in any way into the two houses, and then iterate this step: "Transfer any member with at least two enemies to the other house." CO4,K3

   Let E be the total sum of all the enemies each member has in his own house. In each iterative step,

A. E decreases until it reaches a minimum.

B. E does not change.

C. E increases.

D. E may fluctuate.

4. What is the maximum possible number of kings on an 8×8 chessboard so that no two kings are placed on adjacent – vertically, horizontally, or diagonally – squares. $\boxed{\text{CO4,K2}}$

   A. 16

   B. 32

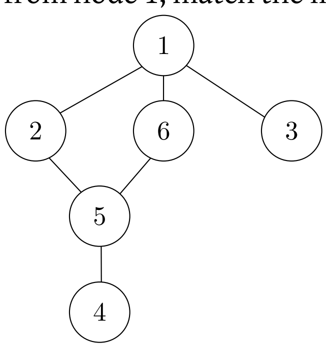
   C. 64

   D. 8

5. Match the graph traversal algorithms with the data structures used in the algorithm to remember the discovered vertices of the graph. $\boxed{\text{CO3,K2}}$

Depth-First Search	Stack
Breadth-First Search	Queue
Best-First Search	Priority Queue
Whatever-First Search	Bag

6. In the graph shown, numbers are priorities of the nodes. The graph is traversed starting from node 1; match the lists on the left side with the correct traversal order on the right side. $\boxed{\text{CO3,K3}}$



1,2,5,4,6,3	depth-first search
1,2,6,3,5,4	breadth-first search
1,2,3,5,4,6	priority-first search (1 highest priority)
1,6,5,4,3,2	priority-first search (1 lowest priority)

7. In whatever-first search outlined below, which of the properties are true for a connected graph? $\boxed{\text{CO2,K3}}$

```
add (_, source) to bag
while bag is not empty
    take (u, v) from bag
    if v is unmarked
        mark v
        parent[v] = u
        for each edge (v,w)
            add (v,w) to bag
```

A. Every vertex of the graph is added to bag at least once

B. Every vertex of the graph is added to bag exactly once

C. All neighbors of a marked vertex are in bag

D. Every vertex of the graph is marked exactly once

8. Recursive backtracking extends a partial solution to a complete solution where the solution is constructed by CO1,K2

   A. an array (fixed number) of choices

   B. a tree of choices

   C. a graph of choices

   D. a list of choices

9. In DFS implemented using a stack, whenever a vertex u is pushed, it is stamped with time `u.start` and whenever it is popped off, it is stamped with time `u.finish`. Suppose vertex w is a child of vertex v. Which of the following are true? CO4,K3

   A. v.start $<$ w.start $<$ w.finish $<$ v.finish

   B. w.start $<$ v.finish $<$ v.start $<$ w.finish

   C. w.start $<$ w.finish $<$ v.start $<$ v.finish

   D. v.start $<$ w.start $<$ v.finish $<$ w.finish

   E. w.start $<$ v.start $<$ w.finish $<$ v.finish

10. What is the space and time required to process all the edges of a graph $G = (V, E)$ represented using standard adjacency lists? CO2,K2

   A. Space = $O(|V| + |E|)$, time = $O(|V| + |E|)$

   B. Space = $O(|V|^2)$, time = $O(|V|^2)$

   C. Space = $O(|V| + |E|)$, time = $O(|V| + |E| \log |E|)$

   D. Space = $O(|V| + |E|)$, time = $O(|V| + \log |E|)$

11. What is the space complexity of recursive backtracking? Let n be the maximum length of the solution sequence, and m the maximum number of choices for each component of the sequence. CO2,K2

    A. $O(n)$

    B. $O(m)$

    C. $O(m+n)$

    D. $O(mn)$

12. What is the time complexity of depth-first search and breadth-first search? Let $|V| = n$ be the number of vertices and $|E| = m$ the number of edges of the graph. CO2,K2

    A. $O(m+n)$ and $O(m+n)$

    B. $O(mn)$ and $O(m+n)$

    C. $O(m+n)$ and $O(n^m)$

    D. $O(m^n)$ and $O(n^m)$

13. Pruning in a backtracking algorithm `Solve(y, k)` means CO4,K2

    A. there is no feasible value for y_{k+1} to extend the partial solution sequence $y_1 \dots y_k$

    B. the solution is not in the subtree rooted at state y_k

    C. the recursive call returns to the parent call `Solve(y, k-1)` to try other choices for y_k

    D. $y_1 \dots y_k$ is a complete solution

14. To find all solutions to 4-queens problem, how many states does backtracking trace without and with pruning? CO2,K3

    A. 65 and 17

    B. 257 and 17

    C. 257 and 65

    D. 65 and 65

15. Match the problems and the problem types. CO3,K2

Problem	Problem Type
Permutations	Combinatorial generation
Hamiltonian cycle	Search
Traveling Sales Person problem	Optimization
Is there a subset of a set that sums to T	Decision

Part B (40 min, 20 marks)

16. Consider the problem of generating all combinations of r things chosen from n things, for some $0 \leq r \leq n$. Without loss of generality, we can assume that the n things are the natural numbers $\{1, 2, \ldots, n\}$. Design a recursive backtracking algorithm as specified below. A combination can be constructed as a sorted sequence of r things. To use the algorithm, call `Choose(y, 0, 1, r, n)`. CO4,K3

1.4.1, 2.1.3

Algorithm: Choose(y, j, k, r, n)

Input: Partial combination $y[1 \ldots j]$ of $j \leq r$ items. Remaining items $k \ldots n$

Output: Combinations $y[1 \ldots r]$ extended from $y[1 \ldots j]$ printed.

a) Trace the state space tree for `Choose(y, 0, 1, 3, 5)`, that generates $\binom{5}{3}$ combinations.

b) Prove that your algorithm is correct.

c) Prove that the combinations generated by this algorithm are in ascending order, that is, when `print(y)` is called, `y[i]` $<$ `y[i+1]` for all $1 \leq i \leq r$.

17. Two men and two women are on one side of a river, and wish to cross the river. They have a boat that can carry at most two people. On any side, women should never be outnumbered by men. CO4,K3

1.4.1, 2.1.3

a) Formulate the problem in state space. Represent the state by a 5-tuple (m_1, w_1, m_2, w_2, b), where m_1 and m_2 are the number of men on the two sides. Similarly, w_1 and w_2. b is the side where the boat is, either L or R. Define a function `next_states(s)` to return the list of the successor states of `s`. Define a function `safe?(s)` that checks whether a state `s` satisfies the constraints of the problem.

b) Design a state space graph search algorithm to find any one safe sequence of river crossings.

c) Draw the state space graph, showing only the safe states.

Part C (20 min, 10 marks)

18. `ColorWFS` is one of the earliest published descriptions of WhateverFirstSearch. Instead of maintaining marked and unmarked vertices, this algorithm maintains a color for each vertex, which is either white, gray, or black. Intuitively, black nodes are "marked" (explored) and gray nodes are "in the bag". Unlike our formulation (in the class) of WhateverFirstSearch, however, `ColorWFS` is not required to process all edges out of a node at the same time. CO4,K3

1.4.1, 2.1.3

Algorithm: ColorWFS (s)

color all nodes white
color s gray
while at least one vertex is gray **do**
    $v \leftarrow$ any gray vertex
    if v has a white neighbor **then**
        $w \leftarrow$ any white neighbor of v
        parent$[w] \leftarrow v$
        color w gray
    else
        color v black
    end
end

a) Prove that ColorWFS maintains the following invariant at all times: No black vertex is a neighbor of a white vertex.
b) Prove that after ColorWFS(s) terminates, all vertices reachable from s are black, all vertices not reachable from s are white, and that the parent edges $v \rightarrow$ parent$[v]$ define a rooted spanning tree (if the given graph is connected).