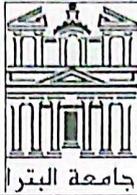


University of Petra		 30 30th Anniversary جامعة البترا - ثلاثون عاما University of Petra
Faculty of Information Technology		كلية تكنولوجيا المعلومات
Department of Computer Science		قسم علوم الحاسوب

**Advanced Algorithms
601326**

Final Exam - 2025/1

Your Name: _____



Instructions for the Exam:

- Write your name and ID number on the exam and answer sheets.
- Write the number of the section that you enrolled in.
- Write the name of your instructor.
- Questions in the exam not allowed.
- Using any type of technology (mobiles, smart watches, etc.) not allowed
- Using extra papers or sheets not allowed

For instructor use only:

Question number	Course ILO	Program ILO	Question weight	Student mark
Q1			5	3
Q2			5	5
Q3			2	zero
Q4	I2		4	1
Q5			5	5
Q6			5	5
Q7	I2		4	0.5
Q8			3	0.5
Q9			7	1
Total			40	21

This exam has 9 Questions. The total mark is 40

3

Question 1) Choose the correct answer for each of the following:

(5 marks)

1. What is time complexity of TSP using Branch and Bound in the worst case? (where n is number of vertices and E is number of edges)

- a) $O(1)$
- b) $O(n \cdot E)$
- c) $O(n^2)$
- d) $O(n!)$

2. What is the space complexity of the Insertion Sort algorithm?

- a) $O(1)$
- b) $O(n^2)$
- c) $O(n)$
- d) $O(n \log n)$

3. Given Build_Max_Heap Algorithm:

```
Build_Max_Heap(A){  
    n= length(A)  
    for i= n/2 down to 1  
        max_heapify(A, i, n)  
}
```

Based on your understanding of the algorithm, why does i start from $n/2$?

- a) to get value of non-leaf vertices
- b) to get index of non-leaf vertices
- c) to divide A into two sub-arrays and sort each one separately
- d) to call `max_heapify` only twice
- e) none of the above

4. Which of the following algorithms design techniques support backtracking:

- a) Divide and conquer
- b) Greedy programming
- c) Dynamic programming
- d) None of the above

5. What is the time complexity of the Dynamic algorithm of 0/1 Knapsack Problem? (where n is number of items, and W is knapsack capacity)

- a) $O(n+W)$
- b) $O(n \cdot W)$
- c) $O(n \log W)$
- d) $O(n^2)$

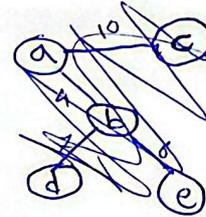
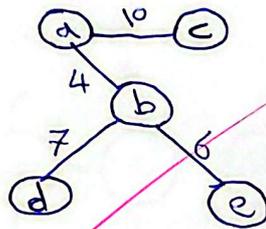
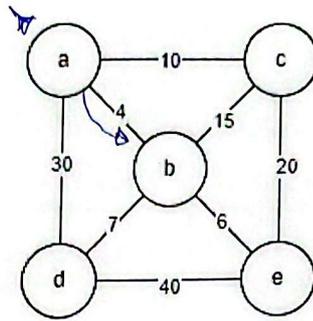
Question 2) Apply Prim's algorithm to the following graph to find MST showing detailed steps. (5 marks)

(5 marks)

5

$h = \{ \cancel{4}, 10, 30, 6, 7, 15 \}$
 $h = \{ \cancel{4}, 7, 10, 15, 30, 20, 40 \}$
 $h = \{ \cancel{4}, \cancel{10}, 15, 20, 30, 40 \}$
 X X X X
 loop

4 (a,b)
 6 (b,e)
 7 (b,d)
 10 (a,c)



Cost = 10 + 4 + 6 + 7 = 27

Question 3) Based on your understanding of Floyd's algorithm (for weighted graphs), write a recursive relation for finding R_{ij}^k . (2 marks)

(2 marks)

2/2

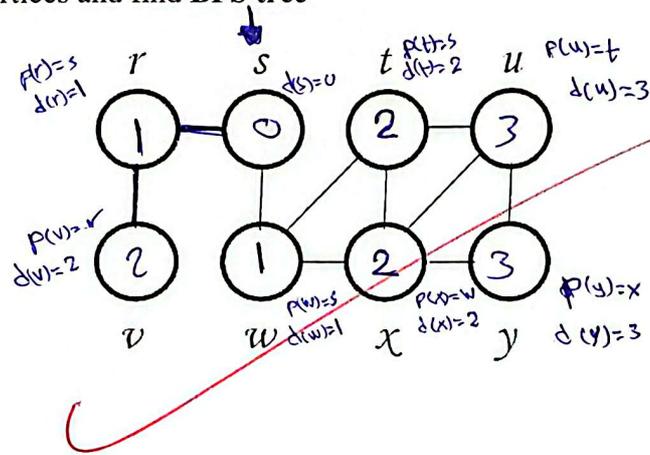
$$T(n) = \begin{cases} 1, & n=1 \\ T(n-1)+1, & n>1 \end{cases}$$

Question 4) Fill in the bellow table best and worst cases of the following algorithms: (4 marks)

BFS

	Breadth-First Search	Quick Sort	Selection Sort	Recursive Towers of Hanoi
Best Case	$O(n)$	$O(n^2)$	$O(n)$	$O(E \log E)$
Worst Case	$O(n^2)$	$O(n^2)$	$O(n^2)$	$O(\frac{E \log E}{2})$ (n^2)

Question 5) Given the below graph, apply the BFS algorithm to find distance from source vertex S to all other vertices and find BFS tree (5 marks)



Question 6) Given the following set of characters and their frequencies:

(5 marks)

Character	A	B	C	D	E	F
Frequency	5	9	12	13	16	45

5

- (a) Construct the Huffman coding tree for the given characters by clearly showing details
 (b) From the constructed tree, determine the binary Huffman code for each character.

A(5) B(9) C(12) D(13) E(16) F(45)

A+B

$$5 + 9 = 14$$

14, C(12), D(13), E(16), F(45)

C+D

$$12 + 13 = 25$$

14, 25, E(16), F(45)

14+E

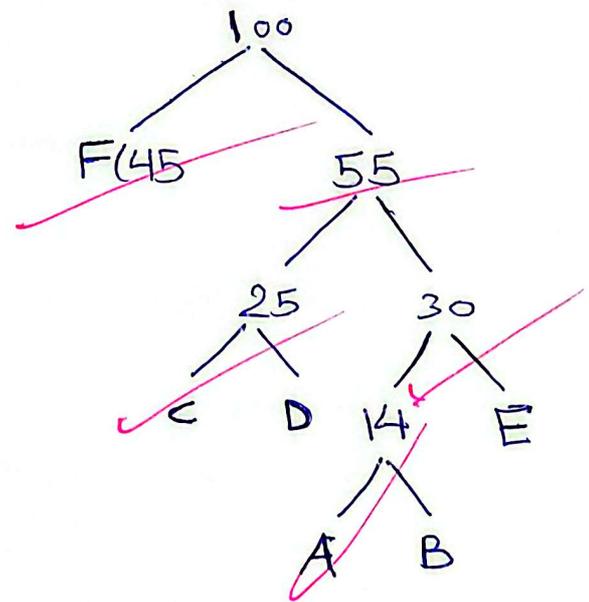
$$14 + 16 = 30$$

25, 30, F(45)

$$25 + 30 = 55$$

F(45), 55

$$F + 55 = 100$$



F = 0

C = 100

D = 101

E = 111

A = 1100

B = 1101

(0.5)

Question 7) Discuss briefly (two sentences maximum) the difference between the following:
(4 marks)

- P vs NP problems:

?

0

- Decision vs Optimization problems:

?

0

- Branch & Bound vs Brute Force techniques:

Branch & Bound :- ~~take~~ take the shortest edge and the smallest cost

Brute Force :- all pairs shortest path problem

0.25

- Traveling Salesman Problem vs Hamiltonian Cycle:

~~take~~ TSP :- take the smallest cost

0.25

Hamiltonian :- ~~take~~ take all pairs from the cost

Question 8)

(3 marks)

Write the recursive formulation for the 0/1 Knapsack Problem for a Dynamic Programming solution.

0.5

let n = number of items
 w = knapsack capacity

w_i = weight of item i
 v_i = value of items

items $n=4$

item 1 :- $w_1=2$, $v_1=3$

item 2 :- $w_2=3$, $v_2=4$

item 3 :- $w_3=4$, $v_3=5$

item 4 :- $w_4=5$, $v_4=6$

	0	1	2	3	4	5
0	0	0	0	0	0	0
1	0	0	3	3	3	3
2	0	0	3	4	4	7
3	0	0	3	4	5	7
4	0	0	3	4	5	7

$$T(n) = \begin{cases} 0, & n=0 \\ 0, & \text{if } w > n \\ v, & \text{if } w < n \end{cases}$$

Question 9)

(7 marks)

Consider the problem of scheduling n jobs of known durations t_1, \dots, t_n for execution by a single processor. The jobs can be executed in any order, one job at a time. You want to find a schedule that minimizes the total time spent by all the jobs in the system. (The time spent by one job in the system is the sum of the time spent by this job in waiting plus the time spent on its execution.)

- a) Design a greedy algorithm for this problem.
- b) Does your greedy algorithm always yield an optimal solution?



~~Kruskal algo
if (n=1) job
else n > job~~

a) Kruskals {
for
if $n = 1$
one job at time
else $n > 1$
at least 2 job or more
total-time = all jobs in the system }

b) No, not always yield an optimal solution

Good Luck