

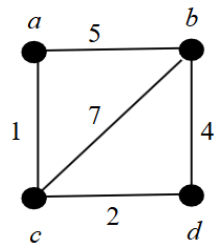
PART I (70 Marks)	
1. Which of the following is used to measure the efficiency of an algorithm?	2.5
[A] Number of lines of its pseudo code [B] The running time of the algorithm on a machine [C] The number of important operations and space used by the algorithm [D] The running time of the algorithm on iphone 6 plus. [E] The size of the algorithm	
2. What is the time complexity of the following algorithm of computing the sum of the first n non-zero natural numbers? input n sum = $n*(n+1)/2$ output sum	2.5
[A] $O(n)$ [B] $O(n^2)$ [C] $O(c)$, where c is a constant. [D] $O(n^3)$ [E] $O(n*(n+1)/2)$	
3. Two algorithms A1, A2 solve a problem with running times of $f_1(n)$ and $f_2(n)$, respectively. Then $f_1(n) \in O(f_2(n))$ means which of the following statements is true	2.5
[A] For all n , $f_1(n) \leq f_2(n)$ [B] There exist n_0 , such that for all $n > n_0$, $f_1(n) \leq f_2(n)$ [C] There exist n_0 and a constant c , such that for all $n > n_0$, $f_1(n) \leq cf_2(n)$ [D] A1 is running fast in all cases. [E] None of the above.	
4. Five algorithms A1, A2, A3, A4, A5 solve a problem with order $f_1(n) = 50\log(\log n) + 20$, $f_2(n) = 10n\log 2n + 100$, $f_3(n) = 10(\log n)^2 + 100$, $f_4(n) = 100n^2 - 3n + 6$, $f_5(n) = n^2/8 - n/4 + 2$, respectively. The algorithm(s) with highest time complexity is (are)	2.5
[A] A1 [B] A2, A3 [C] A4 [D] A4, A5 [E] A5	

Questions 5 to 9 refer to the following algorithm.	
<p>Algorithm: F($A[l \dots r]$)</p> <p>//Input: an array with a position p ($l \leq p \leq r$), such that $A[l] < A[l+1] < \dots < A[p]$ and $A[p] > A[p+1] > \dots > A[r]$.</p> <p>Begin</p> <p> if $l == r$ then</p> <p> return l</p> <p> else</p> <p> $m = \lfloor (l + r) / 2 \rfloor$</p> <p> if $A[m] < A[m + 1]$ then</p> <p> return F($A[m+1 \dots r]$)</p> <p> else</p> <p> return F($A[l, m]$)</p> <p>End</p>	
5. Which algorithm design technique is employed in the above algorithm?	2.5
<p>[A] Brute Force technique</p> <p>[B] Greedy technique</p> <p>[C] Divide- and-Conquer</p> <p>[D] Dynamic Programming</p> <p>[E] Ad hoc technique</p>	
6. The output of the algorithm is	2.5
<p>[A] The largest element in the array</p> <p>[B] A position of the largest element in the array</p> <p>[C] The smallest element in the array</p> <p>[D] A position of the smallest element in the array</p> <p>[E] The element in the middle of the array</p>	
7. What is the number of comparisons to return the output for the input $A[0..7] = [12, 13, 14, 15, 14, 13, 12, 11]$?	2.5
<p>[A] 1</p> <p>[B] 2</p> <p>[C] 3</p> <p>[D] 4</p> <p>[E] 5</p>	

8. If the size n of the array is greater than 1, then the time complexity of the algorithm can be expressed by the recurrence	2.5
[A] $T(n)=2T(n/2) + 1$ [B] $T(n)=2T(n/2) + n$ [C] $T(n)=T(n/2) + n$ [D] $T(n)=T(n/2) + 1$ [E] $T(n)=T(n-1) + T(n-2)$	
9. The time complexity of the algorithm is	2.5
[A] $O(2n)$ [B] $O(\log n)$ [C] $O(2n^2)$ [D] $O(n \log n)$ [E] None of the above	
Questions 10 to 12 refer to the graph G represented by the following adjacency matrix	
$ \begin{array}{c} \\ \\ \\ \\ \\ \\ \\ \\ \end{array} \begin{array}{cccccccc} a & b & c & d & e & f & g & h \\ \begin{array}{l} a \\ b \\ c \\ d \\ e \\ f \\ g \\ h \end{array} \begin{bmatrix} 0 & 1 & 0 & 0 & 1 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 1 & 1 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 \\ 1 & 1 & 1 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 \end{bmatrix} \end{array} $	
10. The total degree of the graph G is	2.5
[A] 21 [B] 18 [C] 19 [D] 20 [E] 10	
11. Starting at the vertex a and resolving ties by the vertex alphabetical order, traverse the graph by breadth-first-search (BFS). Then, the 6 th vertex being visited is	2.5
[A] g [B] h [C] e [D] f [E] c	
12. Starting at the vertex a and resolving ties by the vertex alphabetical order, traverse the graph by depth-first-search (DFS). Then, the last vertex being visited is	2.5

[A] e	
[B] d	
[C] g	
[D] f	
[E] h	
13. Let G be a weighted connected graph	2.5
I. If e is a minimum-weight edge in G , it must be contained in a MST.	
II. If e is a minimum-weight edge in G , it must be contained in each MST.	
III. If e is a maximum-weight edge in G , it must not be contained in any MST.	
Which one of the following is correct?	
[A] I and III are true, II is false	
[B] I and II and III are true	
[C] I and II and III are false	
[D] II and III are true but I is false	
[E] I is true but II and III are false	
14. Let G be a weighted connected graph	2.5
I. If the edge weights are all different G must have exactly one MST	
II. If the edge weights are not all different G must have more than one MST	
III. If the edge weights are all same, every spanning tree of G is a MST	
Which one of the following is correct?	
[A] I is true, II and III are false	
[B] I and III are true but II is false	
[C] I and II and III are true	
[D] I and II and III are false	
[E] II is true but I and III are false	

Note: If a weighted graph is represented by its adjacency matrix, then its element $A[i, j]$ will simply contain the weight of the edge from the i th to the j th vertex if there is such an edge and a special symbol ∞ , if there is no such edge. For example, in the following, the left side is a weighted graph and the right side is its weight matrix



$$\begin{array}{c}
 \begin{array}{ccccc} & a & b & c & d \\
 \begin{array}{c} a \\ b \\ c \\ d \end{array} & \begin{bmatrix} \infty & 5 & 1 & \infty \\ 5 & \infty & 7 & 4 \\ 1 & 7 & \infty & 2 \\ \infty & 4 & 2 & \infty \end{bmatrix}
 \end{array}
 \end{array}$$

Questions 15 to 18 refer to the following weighted graph represented by the following weight matrix:

$$\begin{array}{c}
 \begin{array}{ccccc} & a & b & c & d & e \\
 \begin{array}{c} a \\ b \\ c \\ d \\ e \end{array} & \begin{bmatrix} \infty & 2 & 4 & 4 & 18 \\ 2 & \infty & 4 & 3 & 4 \\ 4 & 4 & \infty & 1 & \infty \\ 4 & 3 & 1 & \infty & 9 \\ 18 & 4 & \infty & 9 & \infty \end{bmatrix}
 \end{array}
 \end{array}$$

15. Let T be a minimum spanning tree of the graph computed using Kruskal's algorithm. The order of edges selected by Kruskal's algorithm is **2.5**

- [A] $(c,d) (a,b) (b,d) (a,c)$
- [B] $(c,d) (a,b) (b,d) (d,e)$
- [C] $(c,d) (a,b) (b,d) (b,e)$
- [D] $(c,d) (a,b) (b,d) (a,d)$
- [E] $(c,d) (a,b) (b,d) (b,c)$

16. Let T be a minimum spanning tree of the graph computed using the Prim's algorithm: Assume vertex a is selected first, then the order of vertices selected by Prim's algorithm is **2.5**

- [A] a, b, d, e, d
- [B] a, b, c, d, e
- [C] a, c, d, b, e
- [D] a, d, c, e, b
- [E] a, b, d, c, e

17. Assume the source vertex is a . Running Dijkstra's algorithm for the graph, after the termination, the label for vertex d is **2.5**

<p>[A] $d(4,b)$,</p> <p>[B] $d(6,b)$,</p> <p>[C] $d(4,a)$,</p> <p>[D] $d(6,a)$,</p> <p>[E] $d(6,c)$,</p>	
18. Assume the source vertex is a . Running Dijkstra's algorithm for the graph, after termination, which one of the following could be an order of vertices selected by Dijkstra's algorithm?	2.5
<p>[A] a, b, e, c, d</p> <p>[B] a, b, d, e, c</p> <p>[C] a, b, d, c, e</p> <p>[D] a, b, e, d, c</p> <p>[E] None of the above</p>	
19. For the three statements below,	2.5
<p>I. A problem in the class P can be solved in worst-case by a polynomial time algorithm.</p> <p>II. A problem in the class NP can be solved by a non-polynomial time algorithms</p> <p>III. A problem in the class NP can be verified in polynomial time</p> <p>Which one of the following is correct?</p>	
<p>[A] I is true, II and III are false</p> <p>[B] I and II are true but III is false</p> <p>[C] I and II are false but III is true</p> <p>[D] II is true but I and III is false</p> <p>[E] None of the above</p>	
20. For the following problems	2.5
<p>I. Vertex Cover Problem.</p> <p>II. Finding minimum spanning tree (MST) in a weighted undirected graph.</p> <p>III. 0/1 Knapsack problem.</p> <p>IV. Traveling Salesman problem.</p> <p>Which one of the following is correct?</p>	
<p>[A] I, II are NP-Complete Problems, III and IV are P-Problems</p> <p>[B] I, II are P-Problems, III and IV are NP-Complete Problems</p> <p>[C] I, III are NP-Complete Problems, II and IV are P-Problems</p> <p>[D] I, III are P-Problems, II and IV are NP-Problems</p> <p>[E] I, III and IV are NP-Complete Problems, II is a P-Problem.</p>	

Questions 21 to 24 refer to the following Longest Common Subsequence problem																																																		
Let $c[i,j]$ be the length of the Longest Common Subsequence of $X_i = x_1, x_2, \dots, x_i$ and $Y_j = y_1, y_2, \dots, y_j$. Then $c[i,j]$ can be recursively defined as following:																																																		
$c[i,j] = \begin{cases} 0 & \text{if } i = 0 \text{ or } j = 0 \\ c[i-1,j-1] + 1 & \text{if } i, j > 0 \text{ and } x_i = y_j \\ \max\{c[i-1,j], c[i,j-1]\} & \text{if } i, j > 0 \text{ and } x_i \neq y_j \end{cases}$																																																		
The following is an incomplete table for the sequences of AATGTT and AGCT.																																																		
<table><tr><td></td><td></td><td>A</td><td>A</td><td>T</td><td>G</td><td>T</td><td>T</td></tr><tr><td></td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td></tr><tr><td>A</td><td>0</td><td></td><td>1</td><td>1</td><td>1</td><td>1</td><td>1</td></tr><tr><td>G</td><td>0</td><td>1</td><td>1</td><td>1</td><td></td><td></td><td></td></tr><tr><td>C</td><td>0</td><td>1</td><td></td><td></td><td></td><td></td><td></td></tr><tr><td>T</td><td>0</td><td>1</td><td>1</td><td></td><td></td><td></td><td></td></tr></table>				A	A	T	G	T	T		0	0	0	0	0	0	0	A	0		1	1	1	1	1	G	0	1	1	1				C	0	1						T	0	1	1					
		A	A	T	G	T	T																																											
	0	0	0	0	0	0	0																																											
A	0		1	1	1	1	1																																											
G	0	1	1	1																																														
C	0	1																																																
T	0	1	1																																															
21. The value of $c[3, 4]$ is		2.5																																																
[A] 1 [B] 2 [C] 3 [D] 4 [E] 5																																																		
22. The length of the longest common subsequence of AATGTT and AGCT is		2.5																																																
[A] 1 [B] 2 [C] 3 [D] 4 [E] 5																																																		
23. The longest common subsequence of AATGT and AGC is		2.5																																																
[A] AGC [B] ATG [C] AAT [D] AGC [E] AG																																																		
24. The longest common subsequence of AATGTT and AGCT is		2.5																																																
[A] AGCT [B] ATGT [C] AATG [D] AGC																																																		

[E] AGT																																														
Questions 25 to 28 refer to the following Knapsack problem: given the following instance of the 0/1 Knapsack problem.																																														
	<table><tr><td>item</td><td>weight</td><td>value</td></tr><tr><td>1</td><td>2</td><td>\$12</td></tr><tr><td>2</td><td>1</td><td>\$10</td></tr><tr><td>3</td><td>3</td><td>\$20</td></tr></table>	item	weight	value	1	2	\$12	2	1	\$10	3	3	\$20																																	
item	weight	value																																												
1	2	\$12																																												
2	1	\$10																																												
3	3	\$20																																												
The Knapsack Capacity W=4																																														
Let V[i, j] be the value of the most valuable subset of the first i items that fit into the Knapsack of capacity j. Then V[i, j] can be recursively defined as follows:																																														
$V[i, j] = \begin{cases} 0 & \text{if } i = 0 \text{ or } j = 0 \\ \max \{V[i - 1, j], \quad v_i + V[i - 1, j - w_i]\} & \text{if } j - w_i \geq 0 \\ V[i - 1, j] & \text{if } j - w_i < 0 \end{cases}$																																														
For the above instance, the following is an incomplete table for V[i, j] (i=0, 1, 2, 3; j=0, 1, 2, 3,4)																																														
	<table><tr><td></td><td></td><td colspan="5">capacity j</td></tr><tr><td>Item</td><td>i</td><td>0</td><td>1</td><td>2</td><td>3</td><td>4</td></tr><tr><td></td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td></tr><tr><td>w₁=2, v₁=12</td><td>1</td><td>0</td><td>0</td><td>12</td><td>12</td><td>12</td></tr><tr><td>w₂=1, v₂=10</td><td>2</td><td>0</td><td>10</td><td>12</td><td>22</td><td>22</td></tr><tr><td>w₃=3, v₃=20</td><td>3</td><td>0</td><td>10</td><td>12</td><td>22</td><td>30</td></tr></table>			capacity j					Item	i	0	1	2	3	4		0	0	0	0	0	0	w ₁ =2, v ₁ =12	1	0	0	12	12	12	w ₂ =1, v ₂ =10	2	0	10	12	22	22	w ₃ =3, v ₃ =20	3	0	10	12	22	30			
		capacity j																																												
Item	i	0	1	2	3	4																																								
	0	0	0	0	0	0																																								
w ₁ =2, v ₁ =12	1	0	0	12	12	12																																								
w ₂ =1, v ₂ =10	2	0	10	12	22	22																																								
w ₃ =3, v ₃ =20	3	0	10	12	22	30																																								
25. The value of V[0, 4] is				2.5																																										
[A] 12 [B] 10 [C] 22 [D] 0 [E] 24																																														
26. What is the value of the most valuable subset that can fit into the knapsack?				2.5																																										
[A] 12 [B] 10 [C] 30 [D] 0 [E] 24																																														
27. Which of the following is an optimal subset of the instance based on the table if the item3 is removed and the capacity of the knapsack is 4?				2.5																																										
[A] {item1, item2} [B] {Item3} [C] {Item1, item2, item3} [D] {Item1, item3}																																														

[E] {Item2, item3}	
28. Which of the following is an optimal subset of the instance based on the table if the capacity of the knapsack is 3 and item3 is removed?	2.5
<p>[A] {item1, item2}</p> <p>[B] {Item3}</p> <p>[C] {Item1, item2, item3}</p> <p>[D] {Item1, item3}</p> <p>[E] {Item2, item3}</p>	

