Sample Question Format

KIIT Deemed to be University

Online Mid Semester Examination(Spring Semester-2021)

Subject Name & Code: **Data Structure and Algorithm & CS-2001**

Applicable to Courses: **B.Tech, Sem-3rd (Regular)**

Full Marks=20 **Time:1 Hour**

SECTION-A(Answer All Questions. All questions carry 2 Marks)

Time:20 Minutes **(5×2=10 Marks)**

<table>
<tr><th>Question No</th><th>Question Type(MCQ/SAT)</th><th>Question</th><th>Answer Key(if MCQ)</th><th>CO Mapping</th></tr>
<tr><td>Q.No:1(a)</td><td></td><td>The following codes create a two dimensional matrix dynamically.
<table><tr><td>Code-A</td><td>Code B</td></tr><tr><td>int *a[4];
a[0]=malloc(12 * sizeof(int));
for(int i=0;i<3;i++)
a[i]=a[0]+(i * 4);</td><td>int *a[4];
for(int i=0;i<3;i++)
a[i]=malloc(4 * sizeof(int));</td></tr></table>Which of the following statement will work properly in both these above codes?
I) a[1][2]=7;
II) a[2]=a[1];
III) *(a[1]+3)=7;
IV) *(a[0]+7)=7;

A) I, II, III
B) I, II, III, IV
C) III, IV
D) I, III, IV</td><td>A</td><td>CO1</td></tr>
<tr><td></td><td></td><td>An array A consists of n integers in locations A[0], A[1] ....A[n-1]. It is required to shift the elements of the array cyclically to the left by k places, where 1 <= k <= (n-1). An incomplete algorithm for doing this in linear time, without using another array is given below. Complete the algorithm by filling in the blanks. Assume</td><td>B</td><td>CO1,CO4</td></tr>
</table>

		all the variables are suitably declared.
		min = n; i = 0;
		while (_____) {
		temp = A[i]; j = i;
		while (_____) {
		A[j] = _____
		j= (j + k) mod n ;
		If (j< min) then
		min =_____ ;
		}
		A[(n + i — k) mod n] =temp
		i = _____
		
		(A) i > min; j!= (n+i+k)mod n; A[(j + k)mod n]; temp; i + 1 ;
		(B) i < min; j!= (n+i-k)mod n; A[(j + k)mod n]; j; i + 1;
		(C) i > min; j!= (n+i+k)mod n; A[(j + k)mod n]; j; i + 1;
		(D) i < min; j!= (n+i-k)mod n; A[(j + k)mod n]; temp; i + 1;		

		Which of the following operations does not take O(1) time for an array of unsorted elements. Assume that array elements are distinct. (A) Find the largest element (B) Delete an element (C) Find the smallest element (D) All of the above	D	CO1,CO2
		A two dimensional array in C is initialized as int A [3][4]. What does *(*(A+3) +2), indicate? A. A[0][3] B. A[1][2] C. A[3][2] D. Compilation Error	D	CO1

Q.No:1(b)		What is the time complexity of following code:
		void function(int n)
		{
		int i, j, k = 0;
		for (i = n; i <= n/2; I--) {
		for (j = n; j <= 2; j = j / 2)
		{
		k = k + n / 2;
		}
		}}
		(a) O(n)
		(b) O(nLogn)
		(c)O(n^2)
		(d) O(n^2Logn)
			b	CO2

		What is the time complexity of following code:
		void fun(int n)
		{
		int a, b;
		for(a=1; a<=n; a=2*a)
		for(b=n/2; b<=n; b++)
		printf("%d%d", a, b);
		}
		
		(a) O (n $\log_2$ n)
		(b) O ($\log_2$ n)
		(c) O (n^2)
		(d) O (n)
			a	CO2

		What is the time complexity of following code:
		void function(int n)
		{
		int i,j,k;
		for(i=n/2; i<=n; i++)
		for(j=1; j + n/2<=n; j= j++)
		for(k=1; k<=n; k= k * 2)
		count++; }
		
		(a) O (n $\log_2$ n)
		(b) O ($n^2\log_2$ n)
		(c) O (n^2)
		(d) O (n)
			b	CO2

		What is the time complexity of following code :
		void function(int n)
		{
		int i, j, k, p, q = 0;
		for (i = 1; i<n; ++i)
		{
		p = 0;
		for (j=n; j>1; j=j/2)
		++p;
		for (k=1; k<p; k=k*2)
		++q;
		}
		
		(a) O (n $\log_2$ n)
		(b) O ($n^2\log_2$ n)
		(c) O (n^2)
		(d) O (n)
			a	CO2

Q.No:1(c)		In a two dimensional matrix A[0...,19, 0..,34] is stored in the memory with each element requiring 4 bytes of storage. If the address of A[0][0] is 2148 and the location of A[k, j] is same in both row-major-order and column-major-order. Find the value of k/j? (a) 34/19 (b)2148/2228 (c) 19/34	a	CO1,CO4

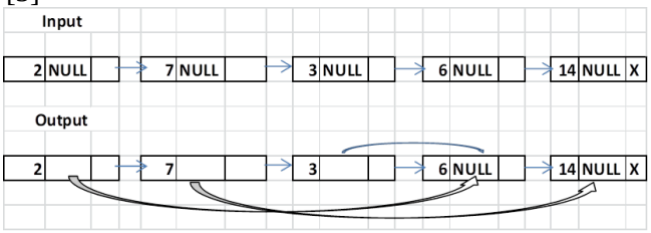
		(d)2228/2148		
		In a two dimensional matrix A[0...,64, 0..,92] is stored in the memory with each element requiring 4 bytes of storage. If the address of A[0][0] is 1000 and the location of A[k, j] is same in both row-major-order and column-major-order. Find the value of k/j? (a) 64/92 (b)1000/2508 (c) 2508/1000 (d) 92/64	d	CO1,CO4
		In a two dimensional matrix A[0...,29, 0..,49] is stored in the memory with each element requiring 4 bytes of storage. If the address of A[0][0] is 1756 and the location of A[k, j] is same in both row-major-order and column-major-order. Find the value of k/j? (a)1756/2234 (b)49/29 (c) 29/49 (d) 2234/1756	b	CO1,CO4
		In a two dimensional matrix A[0...,54, 0..,17] is stored in the memory with each element requiring 4 bytes of storage. If the address of A[0][0] is 7600 and the location of A[k, j] is same in both row-major-order and column-major-order. Find the value of k/j? (a)7600/8540 (b)17/54 (c) 54/17 (d) 8540/7600	b	CO1,CO4
Q.No:1(d)		Evaluate the following postfix expression using stack and indicate the content of the stack when the red marked '*' (2nd multiplication from left to right sequence)is encountered: **5 3 2 * − 18 9 / 4 * 2 / − 6 + 2 –** (a)12, 2, 2 (b) -1, -2 4 (c) -1, 2, 4 (d) 1,-2,2	c	CO4
		Evaluate the following postfix expression using stack and indicate the content of the stack when the operand '+' is encountered: **3 5 10 - 2 + / 5 3 * /** a) 3,5,10 b) 3,-5,2 c) 3,5, 15 d) -5,-2-,3	a	CO4
		Convert the following Infix expression using stack and indicate the content of the stack when the operand '5' is encountered: **((6 + 8) *9 – (5- 4) ^ (2 + 7))**	b	CO4

		a) (- * (- b) (- (c) (-(- d) (*(-		
		Convert the following Infix expression to Postfix using stack and indicate the content of the stack when the operand '5' is encountered: **(1-(2+3)/4)^5+6/7)** a) (-(^ b) (- (^ c) (^ d) ((-/^	c	CO4
Q.No:1(e)		What is the time complexity of the EnQueue and DeQueue operation in a Linear queue? (a) O (1) and O(1) (b) O (n) and) O (n) (c) O (1) and O (n) (d) O (n) and) O (1)	a	CO2
		A normal queue, if implemented using an array of size MAX_SIZE, gets full when: a) Rear=MAX_SIZE-1 b) Front=(rear+1)mod MAX_SIZE c) Front=rear+1 d) Rear=front	a	CO4
		What is the worst case time complexity for a consecutive n EnQueue operation in a linear Queue? (a) O (n $\log_2$ n) (b) O ($n^2\log_2$ n) (c) O (n^2) (d) O (n)	d	CO2
		What is the worst case time complexity for a consecutive n DeQueue operation in a linear Queue? (a) O (n $\log_2$ n) (b) O (n) (c) O ($\log_2$ n) (d) O (n^2)	b	CO2

SECTION-B(Answer Any One Question. Each Question carries 10 Marks)

Time: 30 Minutes **(1×10=10 Marks)**

Question No	Question	CO Mapping
Q.No:2	a. Differentiate between array and linked list. An array has n positive integers. Write a function of O (n) order for removing all the odd numbers from the array. Example, the array contains 10, 2, 3, 7, 8, 6, and 11. The output should be 10, 2, 8, and 6. [5] b. Write the difference between Array and Linked List. Given singly linked list with every node having an	CO1,CO3,CO4

	additional pointer named as 'multiply' that currently points to NULL. Need to make the "Multiply" pointer point to the next multiplied value of the current node. If multiplied value is not present, then keep it NULL. [5] 	
	a. How do we represent a polynomial expression using single linked list? Write a pseudo code to add two polynomial having two numbers of unknown variables. [5] b. Let a linked list consists of n number of nodes, where each node consists of an unique character represents the grades of the students (O, E, A, B, C), and pointer to next node. Write pseudo code/ C code to group the students having same grade in consecutive place and also finally all the nodes should be in sorting order as per their grade value. (O>E>A>B>C) [5]	CO1,CO3,CO4
	a. Write a C program to add two triplets and print the result in another triplet format using the array.[5] b. Write a function in C or Pseudo code: DeleteFromEnd() in a header linked list ,where one Node structure to store an integer value and the special designated node (i.e. header node) contains three information: number of nodes in the list and the maximum, minimum among the list of values. These values must be updated, if required, in every function call. At the beginning define the structure of both the nodes. [5]	CO1,CO3,CO4
Q.No:3	a. Let 'm' number of stacks are implemented in one array where m_i is the size of each i^{th} stack . Write a pseudo code /function for the push () and pop () operations on i^{th} stack.[5] b. Write a function/Pseudo code to swap the following nodes in a circular single Linked List with minimum number of pointers and having only one pointer head/start to indicate first node address. i. 1^{st} node with 2^{nd} node ii. Last node with its previous node (Note: Swap the node/structure node) [5]	CO1,CO4
	a. WAP to find maximum element of stack at a particular instant when any number of push and pop operation is allowed using linked list such that each top node will contain the maximum element from all elements below to it.[5]. b. Design pseudo code/function to add a given value K to each element in the double linked list and if it becomes greater than M, then convert it to 0…M-1 by doing modulo operation with M.(k<M) .Then if the element in the current node is equal to any other node previous to this, delete the current node. [5]	CO1,CO4

	a. Using basic stack push() and pop() operation implement the insert() function as follows. Insert(): Insert function will insert the new element if the element doesn't exist and the insertion will happen using push() such that after insertion the stack elements will be in a sorted manner. [5] b. Write a function to delete all prime numbers present in a doubly linked list. For example, if input: 5->6->11->4->12->16, then output 6->4->12->16. [5]	CO1,CO4

Controller of Examinations