

USN

--	--	--	--	--	--	--	--	--	--

R. V. COLLEGE OF ENGINEERING
Autonomous Institution affiliated to VTU
IV Semester B. E. Fast Track Examinations July-17
Computer Science and Engineering
OPERATING SYSTEMS

*Time: 03 Hours**Maximum Marks: 100**Instructions to candidates:*

1. Answer all questions from Part A. Part A questions should be answered in the first three pages of the answer book only.
2. Answer FIVE full questions from Part B.

PART-A

1	1.1	Mention any two objectives of Operating System.	02
	1.2	List any two controls and status registers.	02
	1.3	Distinguish between preemptive and non-preemptive scheduling.	02
	1.4	Draw the multithreaded process diagram.	02
	1.5	Mention four necessary conditions for dead lock.	02
	1.6	Write functions to implement P and V operations in semaphore.	02
	1.7	With segmentation, if there are 64 segments and maximum segment size is 512 words, then how many bits are used in length of logical address?	02
	1.8	Write the criteria of optimal page replacement algorithm.	02
	1.9	Calculate the average cylinder movements for the following set of references of $FCFS$ scheduling algorithm starting at track 100. 55, 58, 39, 18, 90, 160, 150, 38, 184	02
	1.10	Consider six files F_1, F_2, F_3, F_4, F_5 & F_6 with corresponding sizes 100, 200, 70, 40, 250 & 50 respectively. The files are to be stored on sequential device in such a way as to optimize access time. In what order should the six files be stored?	02

PART-B

2	a	Define spooling. Why is it needed? Explain its working with the necessary diagram.	06
	b	What are the differences between a hard real time system and a soft real time system?	04
	c	List and explain the operating system services.	06
OR			
3	a	With the help of a detailed process state diagram, explain the different states in which a process can be in the system in which a process can be in the system highlighting the different transitions.	06
	b	State and explain any two mechanisms used for inter-process communication.	06
	c	Discuss the benefits of cooperating processes.	04

4	a	Compare and contrast between user level threads and kernel supported threads.	06																																			
	b	With the help of diagrams compare the relative merits and demerits of multithreading models.	06																																			
	c	Distinguish between <i>FCFS</i> and Round Robin scheduling algorithms.	04																																			
OR																																						
5	a	For the table given below, calculate the average turnaround time and the average waiting time for the following algorithms: i) <i>FCFS</i> ii) Preemptive <i>SIF</i> .	10																																			
		<table><tr><th>Process</th><th>Arrival time</th><th>Burst time</th></tr><tr><td>P_1</td><td>0</td><td>8</td></tr><tr><td>P_2</td><td>1</td><td>4</td></tr><tr><td>P_3</td><td>2</td><td>9</td></tr><tr><td>P_4</td><td>3</td><td>5</td></tr></table>		Process	Arrival time	Burst time	P_1	0	8	P_2	1	4	P_3	2	9	P_4	3	5																				
Process	Arrival time	Burst time																																				
P_1	0	8																																				
P_2	1	4																																				
P_3	2	9																																				
P_4	3	5																																				
	b	Explain multiple processor scheduling with an example.	06																																			
6	a	What is a critical section problem? Give a software solution to critical section problem in multiple processes.	08																																			
	b	Explain the synchronizing protocol of a classical readers/writers problem. Write a symbolic program code to implement any one of the above protocol with necessary conditions.	08																																			
OR																																						
7	a	The operating system contains three resources, the number of instance of each resource type are 7,7,10. The current resource allocation state is as shown below: <table><tr><th>Process</th><th colspan="3">Current Allocation</th><th colspan="3">Maximum need</th></tr><tr><th></th><th>R_1</th><th>R_2</th><th>R_3</th><th>R_1</th><th>R_2</th><th>R_3</th></tr><tr><td>P_1</td><td>2</td><td>2</td><td>3</td><td>3</td><td>6</td><td>8</td></tr><tr><td>P_2</td><td>2</td><td>0</td><td>3</td><td>4</td><td>3</td><td>3</td></tr><tr><td>P_3</td><td>1</td><td>2</td><td>4</td><td>3</td><td>4</td><td>4</td></tr></table> i) Is the current allocation in a safe state? ii) Can the request made by process $P_1(1, 1, 0)$ be granted?	Process	Current Allocation			Maximum need				R_1	R_2	R_3	R_1	R_2	R_3	P_1	2	2	3	3	6	8	P_2	2	0	3	4	3	3	P_3	1	2	4	3	4	4	10
	Process	Current Allocation			Maximum need																																	
	R_1	R_2	R_3	R_1	R_2	R_3																																
P_1	2	2	3	3	6	8																																
P_2	2	0	3	4	3	3																																
P_3	1	2	4	3	4	4																																
b	Explain the system model of a deadlock with a neat diagram.																																					
8	a	Discuss internal and external fragmentation with a neat diagram.	06																																			
	b	What are the differences between segmentation and paging?	06																																			
	c	Explain the concept of swapping. Why is it required?	04																																			
OR																																						
9	a	Consider the following page reference. Indicate page faults and calculate total number of page faults for optimal and <i>LRU</i> . The total number of available frames are 1,2,3,2,5,6,3,4,6,3,7,3,1,5,3,6,3,4,2,4,3,4,5,1	08																																			
	b	With the help of a neat diagram, explain the concept of thrashing.	04																																			
	c	What is virtual memory? Explain.	04																																			

10	a	What is disk scheduling? Explain <i>SSTF</i> and <i>SCAN</i> disk scheduling algorithms.	08
	b	What are the various disk management aspects that an operating system is responsible for? Explain them in brief.	08
OR			
11	a	Explain the different file allocation methods bringing out the advantages and disadvantages of each.	08
	b	Discuss the access matrix model of protection and mention the different methods used for the implementation of Access matrix.	08

USN

--	--	--	--	--	--	--	--	--	--

R. V. COLLEGE OF ENGINEERING
Autonomous Institution affiliated to VTU
IV Semester B. E. Fast Track Examinations July-17
Common to CSE / ISE

DESIGN AND ANALYSIS OF ALGORITHMS

Time: 03 Hours

Maximum Marks: 100

Instructions to candidates:

3. Answer all questions from Part A. Part A questions should be answered in the first three pages of the answer book only.
4. Answer FIVE full questions from Part B.

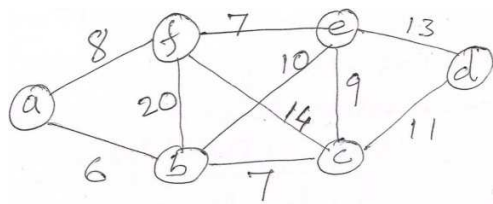
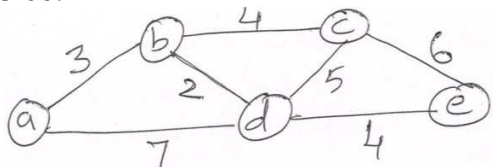
PART-A

1	1.1	What is the best case and worst case time complexity of quick sort?	01
	1.2	Define brute force approach.	01
	1.3	What is the largest number of key comparison required to sort list of n elements using selection sort?	01
	1.4	Why in presorting-based algorithms uses Merge sort for sorting purpose not quick sort?	01
	1.5	What is an AVL tree?	02
	1.6	Differentiate between DFS and BFS algorithms.	02
	1.7	Describe the usage of limits for comparing orders of growth.	02
	1.8	Find out the time complexity of brute-force pattern matching for given string of n characters called text and string of m characters ($m < n$) called pattern.	02
	1.9	Differentiate backtracking and Branch-and-bound approach.	02
	1.10	What is Dijkstra's algorithm? How Dijkstra's algorithm is different from Floyd's algorithm.	02
	1.11	State master theorem, if $f(n) \in \theta(n^d)$ with $d \geq 0$ in recurrence equation.	02
	1.12	Define Decision trees.	02

PART-B

2	a	Discuss the various stages of algorithm design and analysis process using flow charts.	07
	b	Consider the following algorithm: ALGORITHM Enigma ($A[0 \dots n-1; 0 \dots n-1]$) for ($i = 0$ to $n-2$ do) for ($j = i+1$ to $n-1$ do) if ($A[i,j] \neq A[j,i]$) return False; return true	
	i)	What does this algorithm compute?	04
	ii)	Which is the basic operation?	
	iii)	How many times is the basic operation executed?	
	iv)	What is the efficiency class of this algorithm?	

c	<p>Define the mathematical method of comparing the order of growth of two function using limits. Compare order of growth of:</p> <p>i) $n!$ and 2^n</p> <p>ii) $\frac{n(n-1)}{2}$ and n^2</p> <p style="text-align: center;">OR</p> <p>3 a Define each of the asymptotic notation with an example.</p> <p>b Write the general plan for analyzing the time efficiency of recursive algorithm and discuss the time complexity of factorial of the given number problem.</p>	05 08 08															
4 a b c	<p>Design Quicksort algorithm and derive time complexity of Quick sort in best case.</p> <p>Construct AVL tree for the following elements 1,2,3,4,5,6.</p> <p>Describe decrease-and-conquer strategy.</p> <p style="text-align: center;">OR</p> <p>5 a Apply Mergesort to sort the list <i>C,O,M,P,U,T,I,N,G</i> in alphabetical order and also draw tree of calls of Mergesort and tree of calls of Merge.</p> <p>b Solve the topological ordering for the following graph by using DFS method and source removal method.</p> <div style="text-align: center;"> </div> <p>c Explain 2 – 3 tree and construct 2 – 3 tree for the list 9,5,8,3,2,4,7.</p>	08 05 03 06 05 05															
6 a b	<p>Apply Horspool's method to find the pattern <i>BAOBAB</i> in the <i>BESS – KNEW – ABOUT – BAOBAB</i>.</p> <p>Write Warshall's algorithm and explain with following graph.</p> <div style="text-align: center;"> </div> <p>c Design an efficient algorithm to compute binomial coefficient by using dynamic programming.</p> <p style="text-align: center;">OR</p>	05 07 04															
7 a	<p>Write an algorithm for 0/1 Knapsack problem using dynamic programming and solve the following with the same method. Maximum capacity = 5.</p> <table border="1" style="margin-left: auto; margin-right: auto;"> <thead> <tr> <th>Item</th><th>Weight</th><th>Value</th></tr> </thead> <tbody> <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> <tr> <td>4</td><td>2</td><td>15</td></tr> </tbody> </table>	Item	Weight	Value	1	2	12	2	1	10	3	3	20	4	2	15	08
Item	Weight	Value															
1	2	12															
2	1	10															
3	3	20															
4	2	15															

	b	Design Heap sort algorithm and sort the list 2, 9, 7, 6, 5, 8 by Heap sort.	08																									
8	a	Apply prim's algorithm for the following graph and find minimum spanning tree. <div></div>	08																									
	b	Construct a Huffman tree for the following data and obtain its Huffman code: <table><tr><td>Character</td><td>B</td><td>C</td><td>E</td><td>S</td><td>-</td></tr><tr><td>Probability</td><td>0.1</td><td>0.15</td><td>0.15</td><td>0.2</td><td>0.4</td></tr></table> Encode the text BE – CSE using the code. <div>OR</div>	Character	B	C	E	S	-	Probability	0.1	0.15	0.15	0.2	0.4	08													
Character	B	C	E	S	-																							
Probability	0.1	0.15	0.15	0.2	0.4																							
9	a	Write Dijkstra's algorithm and apply the same for the following graph by taking 'a' as source. <div></div>	08																									
	b	Write the decision tree for 3- element Selection sort.	05																									
	c	Describe Huffman tree and write an algorithm for Huffman tree.	03																									
10	a	Explain n-Queen problem and draw state-space tree to solve 4-Queen problem.	07																									
	b	Solve the following Knapsack problem by branch and bound method. The knapsack capacity W is 10. <table><tr><td>Item</td><td>Weight</td><td>Value</td></tr><tr><td>1</td><td>7</td><td>42</td></tr><tr><td>2</td><td>3</td><td>12</td></tr><tr><td>3</td><td>4</td><td>40</td></tr><tr><td>4</td><td>5</td><td>25</td></tr></table> <div>OR</div>	Item	Weight	Value	1	7	42	2	3	12	3	4	40	4	5	25	09										
Item	Weight	Value																										
1	7	42																										
2	3	12																										
3	4	40																										
4	5	25																										
11	a	Write the state-space tree to solve the following instance of the subset sum problem : S = {5, 10, 15, 20, 25}, d = 30.	06																									
	b	Find the optimal solution for the assignment problem given below by brute force method and by branch and bound method. <div><table><tr><td>J₁</td><td>J₂</td><td>J₃</td><td>J₄</td><td></td></tr><tr><td>9</td><td>2</td><td>7</td><td>8</td><td>P₁</td></tr><tr><td>6</td><td>4</td><td>3</td><td>7</td><td>P₂</td></tr><tr><td>5</td><td>8</td><td>1</td><td>8</td><td>P₃</td></tr><tr><td>7</td><td>6</td><td>9</td><td>4</td><td>P₄</td></tr></table></div>	J ₁	J ₂	J ₃	J ₄		9	2	7	8	P ₁	6	4	3	7	P ₂	5	8	1	8	P ₃	7	6	9	4	P ₄	10
J ₁	J ₂	J ₃	J ₄																									
9	2	7	8	P ₁																								
6	4	3	7	P ₂																								
5	8	1	8	P ₃																								
7	6	9	4	P ₄																								

USN

--	--	--	--	--	--	--	--	--	--

R. V. COLLEGE OF ENGINEERING
Autonomous Institution affiliated to VTU
IV Semester B. E. Fast Track Examinations July-17
Computer Science and Engineering
OBJECT ORIENTED PROGRAMMING WITH C++

*Time: 03 Hours**Maximum Marks: 100**Instructions to candidates:*

1. Answer all questions from Part A. Part A questions should be answered in the first three pages of the answer book only.
2. Answer FIVE full questions from Part B.

PART-A

1	1.1	Define namespace.	01
	1.2	Justify why default values must be specified in function prototype and should not be specified again in function definition.	02
	1.3	Write the output for the following program <pre>#include <iostream> using namespace std; Template < class T > void display (const T & a) { cout << a << endl; } Template < class T > void display (const T & a, const int n) { for (int i = 0; i < n; i++) cout << a << endl; } int main () { display ("Welcome to"); display ("COLLEGE", 4); }</pre>	02
	1.4	Explain data abstraction.	02
	1.5	What will be the output of this program? <pre>#include <iostream> using namespace std; int main () { cout << '112'; return 0; }</pre>	01
	1.6	What is stack unwinding?	02
	1.7	Why are insertion and extraction operators overloaded?	01
	1.8	Differentiate between copy constructor and assignment operator.	02

1.9	What is the need to throw class objects instead of fundamental types in exception handling?	02
1.10	Explain rogue pointer.	02
1.11	Differentiate between get () and getline () functions.	02
1.12	Analyze the following code & write the output. <pre>#include <iostream> using namespace std; class Animal { public: Void show () { cout << "\n this is class Animal"; } } class fish { public: void show () { cout << " this is class fish \n"; } } class animal fish: public animal, public fish { }; int main () { animalfish a; a.show (); return 0; }</pre>	02

PART-B

2	a	Write a C++ program to create a class 'Student' and member function accept_data () to accept the details of students like USN, student name and marks in five subjects. Introduce a friend function 'cal_per ()' to calculate and print the percentage of marks.	08
	b	Illustrate with an example, static data members are not part of objects.	04
	c	Explain default arguments whit an example.	04
OR			
3	a	Differentiate between macros and inline functions. Give suitable example.	06
	b	Write a C++ program that cerates objects of class 'point' and initialize its members using overloaded constructors.	08
	c	Analyze and write the output of the following <pre>int main () { int x,y; x = 40; int &ref = x; y = ref; cout << y << endl; y + +; cout << x << ref << y; }</pre>	02

4	a	Depict the role of <code>set_new_handler ()</code> function in overcoming out of memory condition.	04
	b	Justify with an example, why destructor should be used.	04
	C	Write a <code>C++</code> program to allocate memory for an array of integers using <code>new</code> operators and sort all the elements using any sorting technique.	08
OR			
5	a	Derive a class 'withdrawal' from Bank class for withdrawing, derive a 'deposit' class from Bank class for depositing amount into the account and maintain the balance. And now derive a class, Statement, from deposit and withdrawal classes to print the complete bank statement of specific account holder with all transactions. Write an interactive program for <i>N</i> customers using appropriate inheritance mechanism.	10
	b	Illustrate with examples to overcome the ambiguities with diamond shaped inheritance.	06
6	a	Identify the necessity to customize memory allocation by overloading <code>new</code> and <code>delete</code> operators.	05
	b	Justify why should we write our own copy constructor.	05
	c	Show with an example, how to differentiate between overloading of <code>++</code> (increment) operator with respect to prefix and postfix version.	06
OR			
7	a	Write a program to subtract two box type objects by overloading (decrement) " <code>--</code> " operator. Create a class called Box with data members length, breadth and height and member functions as <code>Get_volume ()</code> and <code>Get_values ()</code> .	10
	b	What is a smart pointer? How is it implemented in <code>C++</code> ?	06
8	a	Specify the reason, why computer does not permit creation of instances for an Abstract Base Class. Show with an example.	06
	b	Write a program using <code>C++ STL</code> algorithm – <code>copy ()</code> . Push the data, constructing vectors vector 1 & 2 and print the data. Copy the first few elements of vector 1 into middle of vector 2 and print the result.	10
OR			
9	a	Exemplify the necessity of virtual function.	06
	b	Write a <code>C++</code> program to create a generic function (templates) for quick sort and show the sorting of two data types namely integer and double.	10
10	a	What is the limitation of exception handling and how to overcome the same?	04
	b	Discuss how to insert characters into output streams using <code>write ()</code> to a disk file and to the monitor.	06

	c	Create a file test.txt and write the line "This buffer contains <i>n</i> chars" to an array and print it on the screen also.	06
		OR	
11	a	With suitable examples, discuss the c style error handling and C++ mechanism of error handling.	12
	b	Distinguish between text and binary mode.	04