

Roll No.

2029383

END SEMESTER EXAMINATION DEC - 2025

Name of the Course: B.Tech

Semester: I

Name of the Paper: Fundamental of Computer and Introduction to Programming

Paper Code: TCS101

Time: 3 Hours

Maximum Marks: 100

Note:-

- All questions are compulsory.
- Answer any two sub questions among a, b & c in each main question.
- Q2. is having only two parts. Attempt both the parts.
- Total marks in each main question is twenty.
- Each question carries 10 marks

Q1	(20marks)					
(a)	The Operating System plays a vital role in making a computer usable. Explain in detail the five major functions of the Operating System with real-life examples (e.g., what happens when you open MS Word, play a song, or browse the Internet). Imagine a computer without an Operating System. Describe what tasks would be difficult or impossible to perform and why? Compare how the functions of the OS differ in single-user systems and multi-user systems.	CO1				
(b)	Explain how the transition across the five generations of computers (from vacuum tubes to AI-driven systems) influenced the computer system memory hierarchy design. Provide examples of how memory technologies (e.g., magnetic core, DRAM, cache, SSD) evolved alongside CPU technology and discuss why the memory hierarchy is still essential in modern computers despite advancements in processor speed.					
(c)	Explain various 'C' tokens (datatypes, keywords, constant, identifiers). What is precedence? In which case associativity plays its role. Give suitable example for precedence and associativity.					
Q2	(20 marks)					
(a)	Find output of following codes: <table><tr><td>ii) #include <stdio.h> int main() { int x = 0, y = 5; if(x++ && y++) { printf("Inside IF\n"); } else { printf("Inside ELSE\n"); } printf("x = %d, y = %d", x, y); }</td><td>iii) #include <stdio.h> int main() { int a[5] = {1, 2, 3, 4, 5};</td></tr><tr><td>i) #include <stdio.h> void fun() { static int arr[3] = {1, 2, 3}; int i; for(i = 0; i < 3; i++) arr[i] += 1; printf("%d ", arr[0]); } int main() { fun(); fun(); return 0; }</td><td>iv) #include <stdio.h> int main() { int i, index=0, x;</td></tr></table>	ii) #include <stdio.h> int main() { int x = 0, y = 5; if(x++ && y++) { printf("Inside IF\n"); } else { printf("Inside ELSE\n"); } printf("x = %d, y = %d", x, y); }	iii) #include <stdio.h> int main() { int a[5] = {1, 2, 3, 4, 5};	i) #include <stdio.h> void fun() { static int arr[3] = {1, 2, 3}; int i; for(i = 0; i < 3; i++) arr[i] += 1; printf("%d ", arr[0]); } int main() { fun(); fun(); return 0; }	iv) #include <stdio.h> int main() { int i, index=0, x;	CO2 CO3
ii) #include <stdio.h> int main() { int x = 0, y = 5; if(x++ && y++) { printf("Inside IF\n"); } else { printf("Inside ELSE\n"); } printf("x = %d, y = %d", x, y); }	iii) #include <stdio.h> int main() { int a[5] = {1, 2, 3, 4, 5};					
i) #include <stdio.h> void fun() { static int arr[3] = {1, 2, 3}; int i; for(i = 0; i < 3; i++) arr[i] += 1; printf("%d ", arr[0]); } int main() { fun(); fun(); return 0; }	iv) #include <stdio.h> int main() { int i, index=0, x;					

	<pre> int i = 0; while(i < 5) { printf("%d ", a[i++]); i++; } return 0; } </pre>	<pre> for(i=0; i<10; i++) { int x = 0; while(x++<5) index++; } printf("%d", index); return 0; } </pre>		
	<pre> v) #include<stdio.h> int show() { int p=1; printf("%d\n", p--); } int main() { int i; for(i=0; i<3; i++) show(); return 0; } </pre>			
(b)	<p>Draw a flowchart to print following pattern:</p> <pre> * * ** *** **** ***** </pre>	io bar chart		
Q3	(20 marks)			
(a)	Describe storage classes in 'C' along with their characteristic. Also give suitable example for all storage classes.			
(b)	<p>Write a C Program to calculate the monthly water bill charges by the municipal authority based on the following rules:</p> <p>Base Price:</p> <ul style="list-style-type: none"> Water is charged at 25 paisa per litre. <p>Surcharge on Water Consumption:</p> <ul style="list-style-type: none"> Consumption up to 500 litres → No surcharge Consumption between 501 and 1000 litres → Surcharge of 10 paisa per litre for the amount above 500 litres Consumption above 1000 litres → Surcharge of 12 paisa per litre for the amount above 1000 litres <p>Example Calculation:</p> <ul style="list-style-type: none"> For 1200 litres: <ul style="list-style-type: none"> Base price = 25 paisa × 1200 = Rs. 300 Surcharge: <ul style="list-style-type: none"> 0-500 litres → 0 501-1000 litres → 10 paisa × 500 = Rs. 50 1001-1200 litres → 12 paisa × 200 = Rs. 24 <p>Total Bill = Rs. 300 + Rs. 50 + Rs. 24 = Rs. 374</p>			CO4

(c)	Take a number, reverse its digits and add the reverse to the original. If the sum is not a palindrome, repeat the procedure with the sum until the result is a palindrome. Develop a program in 'C' that takes a number and gives the resulting palindrome and the number of additions it took to find it.							
Q4	(20 marks)							
(a)	Draw and label a neat diagram showing the major segments of memory: <ul style="list-style-type: none">• Code/Text Segment• Data Segment• Stack Segment• Heap Segment Explain the role of each segment in program execution.	CO5						
(b)	develop a 'C' code to pass an integer number into a function <i>count()</i> which return number of digits in that number. If number of digits are even then find sum of digit of the number other wise find multiplication of digits of the number. <table><tr><th>Sample Input</th><th>Sample Output</th></tr><tr><td>Input a number: 234</td><td>Total digits: 3 Sum of digits: 9</td></tr><tr><td>Input a number: 1382</td><td>Total digits: 4 Multiplication of digits: 48</td></tr></table>	Sample Input	Sample Output	Input a number: 234	Total digits: 3 Sum of digits: 9	Input a number: 1382	Total digits: 4 Multiplication of digits: 48	
Sample Input	Sample Output							
Input a number: 234	Total digits: 3 Sum of digits: 9							
Input a number: 1382	Total digits: 4 Multiplication of digits: 48							
(c)	How iteration and recursion are different to each other? Develop a recursive function in 'C' to find sum of following series. $1^2 + 3^2 + 5^2 + 7^2 + 9^2 + \dots + n^2$. Note: n is an odd number.							
Q5	(20 marks)							
(a)	Discuss the following topics related to Arrays: <ul style="list-style-type: none">(a) Declaration and initialization of single-dimensional arrays(b) Computing the address of an element using the base address and the index)(c) Segmentation fault and bounds checking(d) Advantages and limitations of the array	CO6						
(b)	Write a 'C' code which reads <i>n</i> positive integer for an array (where <i>n</i> is an even number). Consider all the elements which are placed at even index as base and element which are placed at odd index as exponent. Create a new array by calculating base raised to power exponent. <table><tr><th>Sample Input</th><th>Sample output</th></tr><tr><td>Number of elements in array: 6 Elements are: 2 4 1 6 5 3</td><td>Elements of new array are: 16 1 125 (2⁴) (1⁶) (5³)</td></tr></table>	Sample Input	Sample output	Number of elements in array: 6 Elements are: 2 4 1 6 5 3	Elements of new array are: 16 1 125 (2 ⁴) (1 ⁶) (5 ³)			
Sample Input	Sample output							
Number of elements in array: 6 Elements are: 2 4 1 6 5 3	Elements of new array are: 16 1 125 (2 ⁴) (1 ⁶) (5 ³)							
(c)	Develop a 'C' code to input <i>n</i> characters in an array. Encrypt the array by replacing each upper-case alphabet with the character '@', each digit with the character '#' and each space with the character '&'. Print the final encrypted array.							