ವಿಶ್ವೇಶ್ವರಯ್ಯ ತಾಂತ್ರಿಕ ವಿಶ್ವವಿದ್ಯಾಲಯ
(ವಿತಾಯುಅಧಿನಿಯಮ್೧೯೯೪ರಅಡಿಯಲ್ಲಿಕರ್ನಾಟಕಸರ್ಕಾರದಿಂದಸ್ಥಾಪಿತವಾದರಾಜ್ಯವಿಶ್ವವಿದ್ಯಾಲಯ)ಜ್ಞಾನ
ಸಂಗಮ" ಮಚ್ಛೆ, ಬೆಳಗಾವಿ-590018

Visvesvaraya Technological University

(The State University of Govt. Karnataka, Established as per VTU Act 1994)
"JnanaSangama" Machhe, Belagavi-590018, www.vtu.ac.in

Dr. A. S. Deshpande B.E., Tech., Ph.D.
Registrar

Phone: (0831) 2498100
Fax: (0831) 2405467

Ref. No. VTU/BGM/BOS/2021-22/ 709 Date: 29 APR 2022.

NOTIFICATION

Subject: Academic Calendar of IV semester MBA, II semester B.Sc., IV semester B.E./ B.Tech., and (revised) VI semester B.E./B.Tech./B. Plan and (revised) I semester B.E./B.Tech./B.Plan/B.Arch. programs of University regarding...

Reference Hon'ble Vice-Chancellor's approval dated: 25.04.2022

Academic Calendar of IV semester MBA, II semester B.Sc., IV semester B.E./ B.Tech., **(revised)** VI semester B.E./B.Tech./B.Plan., and **(revised)** I semester B.E./B.Tech./ B.Plan./ B.Arch., programs of the University are shown on the 2nd page of this notification.

The Principals of Affiliated, Constituent and Autonomous Engineering Colleges are hereby informed to bring the academic calendar to the notice of all concerned.

Sd/-
REGISTRAR

To,

1. The Principals of all affiliated/ constituent /Autonomous Engineering Colleges under the ambit of VTU Belagavi.
2. The chairperson, Department of Mechanical Engineering /Civil Engineering /Computer Science and Engineering and Business Studies of the University.

Copy to.

1. To the Hon'ble Vice-Chancellor through the secretary to VC, VTU Belagavi for information
2. The Registrar (Evaluation), VTU Belagavi for information.
3. The Regional Directors (I/c) of all the regional offices of VTU for circulation.
4. The Director SMUITI, VTU Belagavi for information and to make arrangements to upload Academic Calendar on the VTU web portal.
5. The Director of Physical Education, VTU Belagavi for information
6. PS to Registrar VTU Belagavi
7. All the concerned Special Officer/s and Caseworker/s of the academic section, VTU, Belagavi

Sd/-
REGISTRAR

Academic Calendar for IV sem MBA / IV semB.E./B.Tech.(**Revised**) VI sem B.E./B.Tech /B.Plan., (**Revised**) B.E./B.Tech./B.Arch./B.Plan., and II sem B.Sc. Programs for AY-2021-22

	VI semester B.E./B.Tech. **(Revised)**	VI semester B.Plan. **(Revised)**	IV Semester MBA	IV semester B.E./B.Tech	II semester B.Sc.	I sem B.E./B.Tech./ B.Plan/B.Arch **(Revised)**
Commencement of Semester	04.04.2022	04.04.2022	09.05.2022	16.05.2022	23.05.2022	13.12.2021
Last Working day of Semester	16.07.2022	16.07.2022	20.08.2022	27.08.2022	05.09.2022	10.05.2022
Practical/Viva-Examination	18.07.2022 To 29.07.2022	18.07.2022 To 29.07.2022	----	01.09.2022 To 08.09.2022	06.09.2022 To 09.09.2022	28.05.2022 To 04.06.2022
Theory Examinations	01.08.2022 To 20.08.2022	01.08.2022 To 20.08.2022	22.08.2022 To 14.09.2022	12.09.2022 To 30.09.2022	12.09.2022 To 28.09.2022	12.05.2022 To 27.05.2022
Internship	21.08.2022 To 10.09.2022	21.08.2022 To 10.09.2022	---	---	---	
Internship Viva-Voce/ Project viva	---	---	---	---	---	
Summer Project / Professional training / Organization Study	---	---	----	----	----	
Submission of the report to University	----	---	11.07.2022 To 22.07.2022	---	---	
Commencement of NEXT Semester	19.09.2022	19.09.2022	----	10.10.2022	10.10.2022	06.06.2022

Please Note:

- The academic sessions for EVEN semesters should commence from the **dates** mentioned above.
- All the students of VI semesters B.E./B.Tech. programs have to join the VII semester after completion of their **INTERNSHIP** during the above-mentioned duration.
- **The Institute/Department shall plan to have extra classes to complete the requisite hours of teaching and learning as per the scheme.**
- Faculty should conduct additional tutorial classes in blended mode to solve the doubts of the students.

Faculty should conduct additional tutorial classes in blended mode to solve the doubts of the students.

The faculty/staff shall be available to undertake any work assigned by the university.

Notification regarding the Calendar of Events relating to the conduction of University **Examinations** will be issued by the Registrar (Evaluation) from time to time.

Academic Calendar **may be modified** based on guidelines/directions issued in the future by MHRD/UGC/AICTE/State Government.

Academic Calendar is also applicable for **Autonomous Colleges.** In case any changes are to be effected by Autonomous Colleges in the academic terms and examination schedule, they could do so with the approval of the University.

REGISTRAR

||JAI SRI GURUDEV||

AIT	College Calendar of Events	Format No.	ACD01
		Issue No.	01
		Rev. No.	00

Academic Year : 2021-2022 **Semester :Even**

Sl No.	Date	Event	Remarks
1	4 Apr 2022	Commencement of 6th and 8th semester B.E. classes.	
2	14 Apr 2022	Holiday, Mahaveer Jayanthi, Ambedkar Jayanthi	
3	3 May 2022	Holiday, Basava Jayanthi, Ramzan	
4	21 May 2022	First Test Cycle for 8th semester B.E. Students	
5	23 May 2022	Commencement of 3rd semester B.E. classes.	
6	27 May 2022 to 29 May 2022	First Test Cycle for 6th semester B.E. Students	
7	30 May 2022	Workshop on "Importance of Administrative tasks " for non-teaching staff	
8	6 June 2022	Commencement of 2nd semester B.E. classes.	
9	10 June 2022	Second Test Cycle for 8th semester B.E. Students	
10	14 June 2022	Ethnic Day for 8th semester Students	
11	15 June 2022	Induction Program for first year Students	
12	16 June 2022	Graduation Day for 8th semester Students	
13	17 June 2022 & 18 June 2022	Chunchana-2022	
14	25 June 2022 to 27 June 2022	Second Test Cycle for 6th semester B.E. Students	
15	30 June 2022	Third Test Cycle for 8th semester B.E. Students. Last Working Day for 8th semester B.E. classes.	
16	1 July 2022 to 3 July 2022	First Test Cycle for 4th and 2nd semester B.E. Students	
17	4 July 2022 to 20 July 2022	Theory Examinations for 8th semester B.E. Students	

18	14 July 2022 to 16 July 2022	Third Test Cycle for 6th semester B.E. Students	
19	16 July 2022	Last Working Day for 6th semester B.E. classes.	
20	18 July 2022 to 29 July 2022	Practical Examinations for 6th semester B.E. Students	
21	21 July 2022 to 30 July 2022	Internship viva-voce/Project Viva for 8th Semester Students	
22	31 July 2022 to 2 Aug 2022	Second Test Cycle for 4th and 2nd semester B.E. Students	
23	1 Aug 2022 to 20 Aug 2022	Theory Examinations for 6th semester B.E. Students	
24	9 Aug 2022	Holiday, Muharram	
25	15 Aug 2022	Holiday, Independence Day	
26	27 Aug 2022 to 29 Aug 2022	Third Test Cycle for 4th and 2nd semester B.E. Students	
27	31 Aug 2022	Holiday, Ganesha Chaturthi	
28	1 Sept 2022 to 8 Sept 2022	Practical Examinations for 4th semester B.E. Students	
29	11 Sept 2022 to 29 Sept 2022	Theory Examinations for 2nd semester B.E. Students	
30	12 Sept 2022 to 30 Sept 2022	Theory Examinations for 4th semester B.E. Students	
31	1 Oct 2022 to 10 Oct 2022	Practical Examinations for 2nd semester B.E. Students	

Note: Add any other events like Guest Lecture, National/International Conference ,Seminars, etc in individual department calendar of events .

Copy to:

1. **All HOD's**
2. **Placement Officer**
3. **Establishment Section**
4. **Dhi Team**

PRINCIPAL
AIT, Chickmagalur

DEPARTMENT OF INFORMATION SCIENCE & ENGINEERING

ADICHNUCHANAGIRI INSTITUTE OF TECHNOLOGY, CHIKMAGALUR

Odd Semester

ACADEMIC CALENDAR

2021- 2022 (October 21 to Mar 22)

Week	SUN	MON	TUE	WED	THU	FRI	SAT
W00						Oct 1 Commencement of 5th and 7th Sem	Oct 2 Gandhi Jayanthi Holiday
W01	Oct 3	Oct 4	Oct 5	Oct 6 Mahalaya Amavasya Holiday	Oct 7	Oct 8	Oct 9
W02	Oct 10	Oct 11	Oct12	Oct 13	Oct 14 Ayudha Puja Holiday	Oct 15 Dussehra Holiday	Oct 16
W03	Oct 17	Oct 18 Commencement of 3rd Sem	Oct 19 Ed Milad Holiday	Oct 20 Maharshi Valmiki Jayanthi Holiday	Oct 21	Oct 22	Oct 23
W04	Oct 24	Oct 25	Oct 26	Oct 27	Oct 28	Oct 29	Oct 30
W05	Oct 31	Nov 1 Kannada Rajyotsava Holiday	Nov 2	Nov 3 Naraka Caturdashi Holiday	Nov 4	Nov 5 Balipadyami Holiday	Nov 6
W06	Nov 7	Nov 8 Commencement of 3rd sem MBA	Nov 9	Nov 10	Nov 11 Commencement of 3rd sem MTech	Nov12	Nov 13
W07	Nov 14	Nov 15	Nov 16	Nov 17	Nov 18	Nov 19 IA-1 for 5th and 7th Sem	Nov 20 IA-1 for 5th and 7th Sem
W08	Nov 21 IA-1 for 5th and 7th Sem	Nov 22 Kanakadasa Jayanthi Holiday	Nov 23	Nov 24	Nov 25	Nov 26	Nov 27
W09	Nov 28	Nov 29	Nov 30	Dec 1	Dec 2	Dec 3	Dec 4
W10	Dec 5	Dec 6	Dec 7	Dec 8 IA-1 for 3rd Sem	Dec 9 IA-1 for 3rd Sem	Dec 10 IA-1 for 3rd Sem	Dec 11 Commencement of 1st Sem
W11	Dec 12	Dec 13	Dec 14	Dec 15	Dec 16	Dec 17	Dec 18
W12	Dec 19	Dec 20	Dec 21	Dec 22	Dec 23	Dec 24	Dec 25
W13	Dec 26	Dec 27	Dec 28	Dec 29 IA-2 for 5th and 7th Sem	Dec 30 IA-2 for 5th and 7th Sem	Dec 31 IA-2 for 5th and 7th Sem	Jan 1
W14	Jan 2	Jan 3	Jan 4	Jan 5	Jan 6	Jan 7	Jan 8
W15	Jan 9	Jan 10	Jan 11 IA-2 for 3rd Sem	Jan12 IA-2 for 3rd Sem	Jan 13 IA-2 for 3rd Sem	Jan 14	Jan 15

W16	Jan 16	Jan 17	Jan 18	Jan 19	Jan 20	Jan 21	Jan 22 IA-1 for 1st Sem
W17	Jan 23 IA-1 for 1st Sem	Jan 24 IA-1 for 1st Sem	Jan 25	Jan 26	Jan 27 IA-3 for 5th and 7th Sem	Jan 28 IA-3 for 5th and 7th Sem	Jan 29 IA-3 for 5th and 7th Sem
W18	Jan 30	Jan 31 Last working day for 5th and 7th sem	Feb 1	Feb 2	Feb 3	Feb 4	Feb 5
W19	Feb 6	Feb 7	Feb 8	Feb 9	Feb 10	Feb 11	Feb 12
W20	Feb 13	Feb 14 IA-3 for 3rd Sem	Feb 15 IA-3 for 3rd Sem	Feb 16 IA-3 for 3rd Sem	Feb 17	Feb 18	Feb 19 Last working day for 3rd Sem
W21	Feb 20	Feb 21	Feb 22	Feb 23	Feb 24	Feb 25 IA-2 for 1st Sem	Feb 26 IA-2 for 1st Sem
W22	Feb 27 IA-2 for 1st Sem	Feb 28 Last working day for 3rd sem MBA and M.Tech	Mar 1	Mar 2	Mar 3	Mar 4	Mar 5
W23	Mar 6	Mar 7	Mar 8	Mar 9	Mar 10	Mar 11	Mar 12
W24	Mar 13	Mar 14	Mar 15	Mar 16	Mar 17	Mar 18	Mar 19
W25	Mar 20	Mar 21	Mar 22	Mar 23	Mar 24	Mar 25	Mar 26
W26	Mar 27	Mar 28 IA-3 for 1st Sem	Mar 29 IA-3 for 1st Sem	Mar 30 IA-3 for 1st Sem Last working day 1st Sem			
W27	Apr 3	Apr 4 Commencement Od Even Semester					

NOTE: *1) Practical Examination for V and VII Sem B.E. from 01 Feb to 10 Feb 2022. Theory Examination for V and VII Sem B.E. from 11 Feb to 25 Mar 2022.*
2) Practical Examination for III Sem from 21st Feb to 04 March 2022. Theory Examination for III Sem B.E. from 07 Mar to 25th Mar 2022.
3) Practical Examination for 3rd sem M.Tech from 1st to 5th Mar 2022. Theory Examination for 3rd sem M.Tech from 07 Mar to 25th Mar 2022.
4) Practical Examination for 1st from 1st to 8th Apr 2022. Theory Examination for 11th to 23rd Apr 2022.
5) Commencement of Even Semester VI and VIII SEM from April 4th 2022

Signature of the HOD

Professor & HOD
Department of Computer Science & Engg.,
Adichunchanagiri Institute of Technology,
Chikkamagaluru - 577 102.

DEPARTMENT OF INFORMATION SCIENCE & ENGINEERING

ADICHNUCHANAGIRI INSTITUTE OF TECHNOLOGY, CHIKMAGALUR

Even Semester — ACADEMIC CALENDAR — **2021- 2022(April to September)**

Week	SUN	MON	TUE	WED	THU	FRI	SAT
W00						Apr 1	Apr 2
W01	Apr 3	Apr 4 Commencement of 8th & 6th semester	Apr 5	Apr 6	Apr 7	Apr 8	Apr 9
W02	Apr 10	Apr 11	Apr 12	Apr 13	Apr 14 Mahaveera Jayanthi,Ambedkar Jayanti	Apr 15 Good Friday	Apr 16
W03	Apr 17	Apr 18	Apr 19	Apr 20	Apr 21	Apr 22	Apr 23
W04	Apr 24	Apr 25	Apr 26	Apr 27	Apr 28	Apr 29	Apr 30
W05	May 1	May 2	May 3 Basava Jayanthi, Ramzan	May 4	May 5	May 6	May 7
W06	May 8	May 9	May 10	May 11	May 12	May 13	May 14
W07	May 15	May 16	May 17	May 18	May 19	May 20	May 21 1st Test Cycle for VIII
W08	May 22	May 23 Commencement of 4th semester	May 24	May 25	May 26 1st Technical Seminar	May 27 1st Test Cycle for VI 1st Technical Seminar	May 28 1st Test Cycle for VI
W09	May 29 1st Test Cycle for VI	May 30	May 31	June 1	June 2 Project Seminar-Phase 2	June 3 Project Seminar-Phase 2	June 4
W10	June 5	June 6	June 7 Workshop	June 8	June 9	June 10 2nd Test Cycle for VIII	June 11
W11	June 12	June 13	June 14 Ethnic Day	June 15 Induction Program for 1st year students	June 16 Graduation Day for Final year Students	June 17 Chunchana	June 18 Chunchana
W12	June 19	June 20	June 21	June 22	June 23 2nd Technical Seminar	June 24 2nd Technical Seminar	June 25 2nd Test Cycle for VI
W13	June 26 2nd Test Cycle for VI	June 27 2nd Test Cycle for VI, Project Demo for VIII Semester	June 28 Project Demo for VIII Semester	June 29 Workshop	June 30 3rd Test Cycle for VIII Last Working Day for VIII	July 1 1st Test Cycle for IV, Internship Seminar	July 2 1st Test Cycle for IV
W14	July 3 1st Test Cycle for IV	July 4	July 5	July 6	July 7	July 8	July 9

W15	July 10	July 11	July 12	July 13	July 14	July 15	July 16 Last Working Day for VI
W16	July 17	July 18	July 19	July 20	July 21	July 22	July 23
W17	July 24	July 25	July 26	July 27	July 28	July 29	July 30
W18	July 31 2nd Test Cycle for IV	Aug 1 2nd Test Cycle for IV	Aug 2 2nd Test Cycle for IV	Aug 3	Aug 4	Aug 5	Aug 6
W19	Aug 7	Aug 8	Aug 9 Muharram	Aug 10	Aug 11	Aug 12	Aug 13
W20	Aug 14	Aug 15 Independence Day	Aug 16	Aug 17	Aug 18	Aug 19	Aug 20
W21	Aug 21	Aug 22	Aug 23	Aug 24	Aug 25	Aug 26	Aug 27 3rd Test Cycle for IV
W22	Aug 28 3rd Test Cycle for IV	Aug 29 3rd Test Cycle for IV	Aug 30	Aug 31 Ganesh Chaturthi	Sept 1	Sept 2	Sept 3 Last Working Day for IV

NOTE: ***1) Practical Examination for VI Sem B.E. from 18 July 2022 to 29 July 2022, for 4th Sem: 5th September 2022 to 13th September 2022 & Practical viva exam for VIII sem from 22 July 2022 to 30 July 2022***

2) Theory Examination for VIII Sem B.E. from 4 July 2022 to 20 July 2022 , for VI Sem B.E. from 1 August 2022 to 20 August 2022 & for IV 16th September 2022 to 8th October 2022

3) Commencement of Odd Semester VII SEM 19 September 2022, V SEM 10 October 2022

Signature of the HOD

Professor & HOD
Department of Information Science & Engg.,
Adichunchanagiri Institute of Technology,
Chikkamagaluru - 577 102.

w.E.F 6th June 2022

AIT	Class Timetable	Format No.	ACD06
		Issue No.	01
		Rev. No.	01

Department	IS & E	Semester	IV	Section	
Academic Year	2021-22	Room No.	LH 06		
Class Coordinator	Ashwini Kamath				

Period → / Day ↓ / Time →	1	2	BREAK	3	4	LUNCH	5	6	7
	9.00-10.00	10.00-11.00		11.15-12.15	12.15-1.15		2.30 – 3.20	3.20 – 4.10	4.10-5.00
Monday	DC	MC&ES		DAA	OS		←DAA B1/MP LAB B2→		
Tuesday	M4	OOC		DC	MC&ES		←DAA B2/MP LAB B3→		
Wednesday	OS	DAA		MC&ES	DC		←DAA B3/MP LAB B1→		
Thursday	MC&ES	DAA		OOC	DC		M4	Tutorial class(DAA)	
Friday	OOC	M4		OS	CPC				
Saturday	DAA	OS		M4	OOC				

Allocation of Subjects

Theory

Subject Code	Title	Faculty Name	Faculty Code
18MAT41	Complex Analysis Probability And Statistical Methods	Anitha L + Chitra L S	AL+CLS
18CS42	Design and Analysis of Algorithms	Ashwini Kamath	AK
18CS43	Operating Systems	Deepashri K.S	DKS
18SC44	Microcontroller and Embedded Systems	Anjali B V	ABV
18CS45	Object Oriented Concepts	Suman M H	SMH
18CS46	Data Communication	Rakesh S Raj	RSR
18CPC39	Constitution of India, Professional Ethics and Cyber Law	Divya M N	DMN

Practical:

Subject Code	Title	Faculty Name	Faculty Code
18CSL47	Design and Analysis of Algorithm Laboratory	Ashwini Kamath + Suman M H	AK + SMH
18CSL48	Microprocessor Laboratory	Anjali B V + Deepashri K.S	ABV+DKS

Ashwini Kamath
Signature of the time table coordinator

HOD's Signature
Professor & HOD
Department of Information Science & Engg.
Adichunchanagiri Institute of Technology,
Chikkamagaluru - 577 102.

6 . Course Information

6 . 1 . 1 Course Syllabus

Objectives:
Title of the Course : DESIGN AND ANALYSIS OF ALGORITHMS
Subject Code : 18CS42

Module 1

Introduction :

What is an Algorithm?, Algorithm Specification, Analysis Framework, Performance Analysis- Space complexity, Time complexity , Asymptotic Notations- Big-Oh notation (O), Omega notation (#), Theta notation (#), and Little-oh notation (o), Mathematical analysis of Non-Recursive and recursive Algorithms with Examples , Important Problem Types-Sorting, Searching, String processing, Graph Problems, Combinatorial Problems, Fundamental Data Structures-Stacks, Queues, Graphs, Trees, Sets and Dictionaries

Module 2

Divide and Conquer :

General method, Binary search, Recurrence equation for divide and conquer, Finding the maximum and minimum , Merge sort, Quick sort , Strassen's matrix multiplication, Advantages and Disadvantages of divide and conquer, Decrease and Conquer Approach- Topological Sort

Module 3

Greedy Method :

General method, Coin Change Problem, Knapsack Problem, Job sequencing with deadlines, Minimum cost spanning trees- Prim's Algorithm, Kruskal's Algorithm , Single source shortest paths- Dijkstra's Algorithm, Optimal Tree problem-Huffman Trees and Codes , Transform and Conquer Approach- Heaps and Heap Sort

Module 4

Dynamic Programming :

General method with Examples, Multistage Graphs , Transitive Closure- Warshall's Algorithm, All Pairs Shortest Paths-Floyd's Algorithm, Optimal Binary Search Trees, Knapsack problem , Bellman-Ford Algorithm , Travelling Sales Person problem , Reliability design

Module 5

Backtracking :

General method, N-Queens problem , Sum of subsets problem , Graph coloring , Hamiltonian cycles, Programme and Bound- Assignment Problem, Travelling Sales Person problem, 0/1 Knapsack problem- LC Programme and Bound solution , FIFO Programme and Bound solution, NP-Complete and NP-Hard problems: Basic concepts, non-10 deterministic algorithms, P, NP, NP-Complete, and NP-Hard classes

6 . Course Information

6 . 1 . 2 Text Books and Reference Books

TEXT BOOKS :

1 . Introduction to the Design and Analysis of Algorithms, Anany Levitin:, 2rd Edition, 2009. Pearson

2 . Computer Algorithms/C++, Ellis Horowitz, Satraj Sahni and Rajasekaran, 2nd Edition, 2014, Universities Press

REFERENCE BOOKS :

1 . Introduction to Algorithms, Thomas H. Cormen, Charles E. Leiserson, Ronal L. Rivest, Clifford Stein, 3rd Edition, PHI

2 . Design and Analysis of Algorithms , S. Sridhar, Oxford (Higher Education).

HOD

6 . Course Information

6 . 2

Semester : 4 **Section :** A **Course :** DESIGN AND ANALYSIS OF ALGORITHMS

Period	Planned			Execution		
	Date	Topic	Source material to be referred	Date	Topic	Source material to be referred
			1			
1	2022-05-23	What is an Algorithm?, Algorithm Specification	Text 2	2022-05-23	What is an Algorithm?, Algorithm Specification	Text 2
2	2022-05-25	Analysis Framework,Performance Analysis- Space complexity	Text 1, Text 2,	2022-05-25	Analysis Framework, Performance Analysis- Space complexity	Text 1, Text 2,
3	2022-05-26	Time complexity , Asymptotic Notations- Big-Oh notation (O)	Text 1, Text 2,	2022-05-26	Time complexity, Asymptotic Notations- Big-Oh notation (O)	Text 1, Text 2,
4	2022-05-28	Omega notation (#), Theta notation (#)	Text 1	2022-05-28	Omega notation (#), Theta notation (#)	Text 1
5	2022-05-30	and Little-oh notation (o), Mathematical analysis of Non-Recursive and recursive Algorithms with Examples	Text 1	2022-05-30	and Little-oh notation (o), Mathematical analysis of Non-Recursive and recursive Algorithms with Examples	Text 1
6	2022-06-01	Mathematical analysis of Non-Recursive and recursive Algorithms with Examples	Text 1	2022-06-01	Mathematical analysis of Non-Recursive and recursive Algorithms with Examples	Text 1
7	2022-06-02	Mathematical analysis of Non-Recursive and recursive Algorithms with Examples	Text 1	2022-06-06	Mathematical analysis of Non-Recursive and recursive Algorithms with Examples	Text 1
8	2022-06-04	Important Problem Types- Sorting, Searching, Graph Problems, String processing, Combinatorial Problems	Text 1	2022-06-08	Important Problem Types- Sorting, Searching, Graph Problems, String processing, Combinatorial Problems	Text 1
9	2022-06-06	Fundamental Data Structures-Stacks, Queues	Text 1	2022-06-09	Fundamental Data Structures-Stacks, Queues	Text 1
10	2022-06-08	Graphs, Trees, Sets and Dictionaries	Text 1	2022-06-09	Graphs, Trees, Sets and Dictionaries	Text 1
11	2022-06-09	What is an Algorithm?, Omega notation (#), Asymptotic Notations- Big-Oh notation (O), Theta notation (#)	Text 1, Text 2,	2022-06-10	What is an Algorithm?, Omega notation (#), Asymptotic Notations- Big-Oh notation (O), Theta notation (#)	Text 1, Text 2,

12	2022-06-09	What is an Algorithm?, Performance Analysis- Space complexity, Time complexity, Asymptotic Notations- Big-Oh notation (O), Omega notation (#), Theta notation (#)	Text 1, Text 2,	2022-06-11	What is an Algorithm?, Performance Analysis- Space complexity, Time complexity, Asymptotic Notations- Big-Oh notation (O), Omega notation (#), Theta notation (#)	Text 1, Text 2,
			2			
13	2022-06-11	General method	Text 2	2022-06-13	General method	Text 2
14	2022-06-13	Binary search	Text 2	2022-06-15	Binary search	Text 2
15	2022-06-15	Binary search	Text 2	2022-06-16	Binary search	Text 2
16	2022-06-16	Recurrence equation for divide and conquer	Text 2	2022-06-20	Recurrence equation for divide and conquer	Text 2
17	2022-06-18	Finding the maximum and minimum	Text 2	2022-06-22	Finding the maximum and minimum	Text 2
18	2022-06-20	Merge sort	Text 1, Text 2,	2022-06-23	Merge sort	Text 1, Text 2,
19	2022-06-22	Quick sort	Text 1, Text 2,	2022-06-23	Quick sort	Text 1, Text 2,
20	2022-06-23	Strassen's matrix multiplication	Text 1	2022-06-25	Strassen's matrix multiplication	Text 1
21	2022-06-23	Merge sort, Quick sort	Text 1, Text 2,	2022-06-27	Merge sort, Quick sort	Text 1, Text 2,
22	2022-06-25	Advantages and Disadvantages of divide and conquer	Text 1, Text 2,	2022-06-29	Advantages and Disadvantages of divide and conquer	Text 1, Text 2,
23	2022-06-27	Decrease and Conquer Approach- Topological Sort	Text 1	2022-06-30	Decrease and Conquer Approach- Topological Sort	Text 1
			3			
24	2022-06-29	General method	Text 2	2022-07-04	General method	Text 2
25	2022-06-30	Coin Change Problem	Text 2	2022-07-06	Coin Change Problem	Text 2
26	2022-07-04	Knapsack Problem	Text 1, Text 2,	2022-07-07	Knapsack Problem	Text 1, Text 2,
27	2022-07-06	Knapsack Problem	Text 1, Text 2,	2022-07-07	Knapsack Problem	Text 1, Text 2,
28	2022-07-07	Job sequencing with deadlines	Text 2	2022-07-09	Job sequencing with deadlines	Text 2
29	2022-07-07	Job sequencing with deadlines	Text 2	2022-07-11	Job sequencing with deadlines	Text 2
30	2022-07-09	Minimum cost spanning trees- Prim's Algorithm	Text 1	2022-07-16	Minimum cost spanning trees- Prim's Algorithm	Text 1
31	2022-07-11	Kruskal's Algorithm	Text 1, Text 2,	2022-07-18	Kruskal's Algorithm	Text 1, Text 2,
32	2022-07-13	Kruskal's Algorithm	Text 1, Text 2,	2022-07-20	Kruskal's Algorithm	Text 1, Text 2,
33	2022-07-14	Single source shortest paths- Dijkstra's Algorithm	Text 1	2022-07-21	Single source shortest paths- Dijkstra's Algorithm	Text 1
34	2022-07-16	Optimal Tree problem- Huffman Trees and Codes	Text 1, Text 2,	2022-07-21	Optimal Tree problem- Huffman Trees and Codes	Text 1, Text 2,

35	2022-07-18	Transform and Conquer Approach- Heaps and Heap Sort	Text 1	2022-07-23	Transform and Conquer Approach- Heaps and Heap Sort	Text 1
			4			
36	2022-07-20	General method with Examples	Text 2	2022-07-25	General method with Examples	Text 2
37	2022-07-21	Multistage Graphs	Text 2	2022-07-27	Multistage Graphs	Text 2
38	2022-07-21	Multistage Graphs	Text 2	2022-07-28	Multistage Graphs	Text 2
39	2022-07-23	Transitive Closure- Warshall's Algorithm	Text 1	2022-08-03	Transitive Closure- Warshall's Algorithm	Text 1
40	2022-07-25	All Pairs Shortest Paths- Floyd's Algorithm	Text 1	2022-08-04	All Pairs Shortest Paths- Floyd's Algorithm	Text 1
41	2022-07-27	Optimal Binary Search Trees	Text 1	2022-08-04	Optimal Binary Search Trees	Text 1
42	2022-07-28	Knapsack problem	Text 2, Text 1,	2022-08-05	Knapsack problem	Text 2, Text 1,
43	2022-08-03	Bellman-Ford Algorithm	Text 1, Text 2,	2022-08-06	Bellman-Ford Algorithm	Text 1, Text 2,
44	2022-08-04	Travelling Sales Person problem	Text 2	2022-08-08	Travelling Sales Person problem	Text 2
45	2022-08-04	Optimal Binary Search Trees, Travelling Sales Person problem	Text 1, Text 2,	2022-08-10	Optimal Binary Search Trees, Travelling Sales Person problem	Text 1, Text 2,
46	2022-08-06	Reliability design	Text 2	2022-08-11	Reliability design	Text 2
			5			
47	2022-08-08	General method, N-Queens problem	Text 1, Text 2,	2022-08-11	General method, N-Queens problem	Text 1, Text 2,
48	2022-08-10	Sum of subsets problem , Graph coloring	Text 1	2022-08-12	Sum of subsets problem, Graph coloring	Text 1
49	2022-08-11	Hamiltonian cycles, Programme and Bound- Assignment Problem	Text 1	2022-08-13	Hamiltonian cycles, Programme and Bound- Assignment Problem	Text 1
50	2022-08-13	Travelling Sales Person problem	Text 1	2022-08-17	Travelling Sales Person problem	Text 1
51	2022-08-17	0/1 Knapsack problem- LC Programme and Bound solution	Text 1	2022-08-18	0/1 Knapsack problem- LC Programme and Bound solution	Text 1
52	2022-08-18	FIFO Programme and Bound solution	Text 1, Text 2,	2022-08-20	FIFO Programme and Bound solution	Text 1, Text 2,
53	2022-08-20	NP-Complete and NP-Hard problems: Basic concepts	Text 1	2022-08-22	NP-Complete and NP-Hard problems: Basic concepts	Text 1
54	2022-08-22	non-10 deterministic algorithms	Text 1	2022-08-24	non-10 deterministic algorithms	Text 1
55	2022-08-24	P	Text 1	2022-08-25	P	Text 1
56	2022-08-25	NP	Text 1	2022-08-25	NP	Text 1
57	2022-09-01	NP-Complete	Text 1	2022-09-01	NP-Complete	Text 1
58	2022-09-03	and NP-Hard classes	Text 1	2022-09-03	and NP-Hard classes	Text 1

||Jai Sri Gurudev ||

Adichunchanagiri Shikshana Trust (R)

Adichunchanagiri Institute of Technology, Chikmagalur - 577102

Date: 04-05-2022

Internal Test Schedule for Eighth Semester

1st Test Cycle

DATE	21-05-2022	21-05-2022
DAY	Saturday	Saturday
TIME	09.00 AM to 10.00 AM	12.00 PM to 01.00 PM
Subject Code	18XX81	18XX82X

2nd Test Cycle

DATE	10-06-2022	10-06-2022
DAY	Friday	Friday
TIME	09.00 AM to 10.00 AM	12.00 PM to 01.00 PM
Subject Code	18XX81	18XX82X

3rd Test Cycle

DATE	30-06-2022	30-06-2022
DAY	Thursday	Thursday
TIME	09.00 AM to 10.00 AM	12.00 PM to 01.00 PM
Subject Code	18XX81	18XX82X

Tech Co-ordinators: RPM

Dr. C. T. JAYADEVA
Principal
B.E.,M.Tech.,Ph.D.
Adichunchanagiri Institute of Technology
CHIKKAMAGALURU-577102

Adichunchanagiri Shikshana Trust (R)

Adichunchanagiri Institute of Technology, Chikmagalur - 577102

Date: 04-05-2022

Internal Test Schedule for Sixth Semester

1st Test Cycle

DATE	27-05-2022	27-05-2022	28-05-2022	28-05-2022	29-05-2022
DAY	Friday	Friday	Saturday	Saturday	Sunday
TIME	09.00 AM to 10.00 AM	12.00 PM to 01.00 PM	09.00 AM to 10.00 AM	12.00 PM to 01.00 PM	09.00 AM to 10.00 AM
Subject Code	18XX61	18XX62	18XX63	18XX64X	18XX65X

2nd Test Cycle

DATE	25-06-2022	25-06-2022	26-06-2022	26-06-2022	27-06-2022
DAY	Saturday	Saturday	Sunday	Sunday	Monday
TIME	09.00 AM to 10.00 AM	12.00 PM to 01.00 PM	09.00 AM to 10.00 AM	12.00 PM to 01.00 PM	09.00 AM to 10.00 AM
Subject Code	18XX61	18XX62	18XX63	18XX64X	18XX65X

3rd Test Cycle

DATE	14-07-2022	14-07-2022	15-07-2022	15-07-2022	16-07-2022
DAY	Thursday	Thursday	Friday	Friday	Saturday
TIME	09.00 AM to 10.00 AM	12.00 PM to 01.00 PM	09.00 AM to 10.00 AM	12.00 PM to 01.00 PM	09.00 AM to 10.00 AM
Subject Code	18XX61	18XX62	18XX63	18XX64X	18XX65X

NB

Test Co-ordinator: Mr RSM

Dr. C. T. JAYADEVA
Principal
Adichunchanagiri Institute of Technolo
CHIKKAMAGALURU-577102

||Jai Sri Gurudev ||

Adichunchanagiri Shikshana Trust (R)

Adichunchanagiri Institute of Technology, Chikmagalur - 577102

Internal Test Schedule for Fourth Semester

Date: 02-06-2022

1st Test Cycle

DATE	01-07-2022	01-07-2022	01-07-2022	02-07-2022	02-07-2022	03-07-2022	03-07-2022
DAY	Friday	Friday	Friday	Saturday	Saturday	Sunday	Sunday
TIME	09.00 AM to 10.00 AM	12.00 PM to 01.00 PM	03.30 PM to 04.30 PM	09.00 AM to 10.00 AM	12.00 PM to 01.00 PM	09.00 AM to 10.00 AM	12.00 PM to 01.00 PM
Subject Code	18MAT41	18XX42	18KVK/KAK/CPC 49	18XX43	18XX44	18XX45	18XX46

2nd Test Cycle

DATE	31-07-2022	31-07-2022	01-08-2022	01-08-2022	02-08-2022	02-08-2022	02-08-2022
DAY	Sunday	Sunday	Monday	Monday	Tuesday	Tuesday	Tuesday
TIME	09.00 AM to 10.00 AM	12.00 PM to 01.00 PM	09.00 AM to 10.00 AM	12.00 PM to 01.00 PM	09.00 AM to 10.00 AM	12.00 PM to 01.00 PM	03.30 PM to 04.30 PM
Subject Code	18MAT41	18XX42	18XX43	18XX44	18XX45	18XX46	18KVK/KAK/CPC 49

3rd Test Cycle

DATE	27-08-2022	27-08-2022	28-08-2022	28-08-2022	29-08-2022	29-08-2022	29-08-2022
DAY	Saturday	Saturday	Sunday	Sunday	Monday	Monday	Monday
TIME	09.00 AM to 10.00 AM	12.00 PM to 01.00 PM	09.00 AM to 10.00 AM	12.00 PM to 01.00 PM	09.00 AM to 10.00 AM	12.00 PM to 01.00 PM	03.30 PM to 04.30 PM
Subject Code	18MAT41	18XX42	18XX43	18XX44	18XX45	18XX46	18KVK/KAK/CPC 49

Dr. C.T. JAYADEVA
Principal B.E.,M.Tech.,Ph.D.
Principal
Adichunchanagiri Institute of Technology
CHIKKAMAGALURU-577102

||Jai Sri Gurudev ||

Adichunchanagiri Shikshana Trust (R)

Adichunchanagiri Institute of Technology, Chikmagalur - 577102

Internal Test Schedule for Second Semester Physics Cycle

Date: 09-06-2022

1st Test Cycle

DATE	01-07-2022	01-07-2022	02-07-2022	02-07-2022
DAY	Friday	Friday	Saturday	Saturday
TIME	09.00 AM to 10.00 AM	12.00 PM to 01.00 PM	09.00 AM to 10.00 AM	12.00 PM to 01.00 PM
Subject Code	21MAT21	21PHY22	21ELE23	21CIV24

2nd Test Cycle

DATE	31-07-2022	31-07-2022	01-08-2022	01-08-2022
DAY	Sunday	Sunday	Monday	Monday
TIME	09.00 AM to 10.00 AM	12.00 PM to 01.00 PM	09.00 AM to 10.00 AM	12.00 PM to 01.00 PM
Subject Code	21MAT21	21PHY22	21ELE23	21CIV24

3rd Test Cycle

DATE	27-08-2022	27-08-2022	28-08-2022	28-08-2022
DAY	Saturday	Saturday	Sunday	Sunday
TIME	09.00 AM to 10.00 AM	12.00 PM to 01.00 PM	09.00 AM to 10.00 AM	12.00 PM to 01.00 PM
Subject Code	21MAT21	21PHY22	21ELE23	21CIV24

Dr. C. T. JAYADEVA
Principal
B.E., M.Tech., Ph.D.
Adichunchanagiri Institute of Technology
CHIKKAMAGALURU-577102

||Jai Sri Gurudev ||

Adichunchanagiri Shikshana Trust (R)

Adichunchanagiri Institute of Technology, Chikmagalur - 577102

Internal Test Schedule for Second Semester Chemistry Cycle

Date: 09-06-2022

1st Test Cycle

DATE	01-07-2022	01-07-2022	02-07-2022	02-07-2022	03-07-2022
DAY	Friday	Friday	Saturday	Saturday	Sunday
TIME	09.00 AM to 10.00 AM	12.00 PM to 01.00 PM	09.00 AM to 10.00 AM	12.00 PM to 01.00 PM	09.00 AM to 10.00 AM
Subject Code	21MAT21	21PSP23	21CHE22	21ELN24	21EME25

2nd Test Cycle

DATE	31-07-2022	31-07-2022	01-08-2022	01-08-2022	02-08-2022
DAY	Sunday	Sunday	Monday	Monday	Tuesday
TIME	09.00 AM to 10.00 AM	12.00 PM to 01.00 PM	09.00 AM to 10.00 AM	12.00 PM to 01.00 PM	09.00 AM to 10.00 AM
Subject Code	21MAT21	21PSP23	21CHE22	21ELN24	21EME25

3rd Test Cycle

DATE	27-08-2022	27-08-2022	28-08-2022	28-08-2022	29-08-2022
DAY	Saturday	Saturday	Sunday	Sunday	Monday
TIME	09.00 AM to 10.00 AM	12.00 PM to 01.00 PM	09.00 AM to 10.00 AM	12.00 PM to 01.00 PM	09.00 AM to 10.00 AM
Subject Code	21MAT21	21PSP23	21CHE22	21ELN24	21EME25

Dr. C. T. JAYADEVA
Principal B.E.,M.Tech.,Ph.D.
Adichunchanagiri Institute of Technology
CHIKKAMAGALURU-577102

USN : _____

Adichunchanagiri Institute of Technology
DEPARTMENT OF INFORMATION SCIENCE AND ENGINEERING
I - INTERNAL ASSESSMENT

Semester: 4-CBCS 2018 Date: 01 Jul 2022
Subject: DESIGN AND ANALYSIS OF ALGORITHMS (18CS42) Time: 12:00 PM - 01:00 PM
Faculty: Ms Ashwini Kamath Max Marks: 50

Answer all questions

Marks CO PO BT/CL

1. Define algorithm. Discuss the criteria's that an algorithm must satisfy with an example algorithm.

[10.0] 1 [1, 2, 3, 4, 9, 10, 12] [1]

OR

2. List and define various basic efficiency classes.

[10.0] 1 [1, 2, 3, 4, 9, 10, 12] [1]

3. Explain asymptotic notations BigO, BigΩ and Bigθ that are used to compare the order of growth of an algorithm with example.

[10.0] 1 [1, 2, 3, 4, 9, 10, 12] [2]

OR

4. Explain general plan of mathematical analysis ofnon-recursive

Algorithms with example.

[10.0] 1 [1, 2, 3, 4, 9, 10, 12] [2]

5. Define best-case, worst-case and average-case efficiency. Write the algorithm and give these efficiencies for sequential search.

[10.0] 2 [1, 2, 3, 4, 9, 10, 12] [3]

OR

6. Write the algorithm to findmaximum elementin the given array and explain the mathematical analysis of this non-recursive algorithm.

[10.0] 2 [1, 2, 3, 4, 9, 10, 12] [3]

7. Write the algorithm to check whether all the elements in the given array aredistinctand explain the mathematical analysis of this non- recursive algorithm. Derive its worst-case time complexity

[10.0] 2 [1, 2, 3, 4, 9, 10, 12] [3]

OR

8. Write the algorithm to performmatrix multiplicationand explain the mathematical analysis of this non-recursive algorithm

[10.0] 2 [1, 2, 3, 4, 9, 10, 12] [3]

9. write the recursive algorithm to find theFactorialof a given number.

and illustrate its mathematical analysis.

[10.0] 2 [1, 2, 3, 4, 9, 10, 12] [3]

OR

10. Illustrate mathematical analysis of recursive algorithm forTowers ofHanoi.

[10.0] 2 [1, 2, 3, 4, 9, 10, 12] [3]

Adichunchanagiri Institute of Technology
DEPARTMENT OF INFORMATION SCIENCE AND ENGINEERING
I - INTERNAL ASSESSMENT Scheme

Semester: 4-CBCS 2018
Subject: DESIGN AND ANALYSIS OF ALGORITHMS (18CS42)
Faculty: Ms Ashwini Kamath

Date: 01 Jul 2022
Time: 12:00 PM - 01:00 PM
Max Marks: 50

Answer all questions

1. Define algorithm. Discuss the criteria's that an algorithm must satisfy with an example algorithm. [6+4=-10 Marks]

- An algorithm is a finite set of instructions that, if followed , accomplishes a particular task. In addition , all algorithms must satisfy the following criteria:
- Input : Zero or more quantities are externally supplied.
- Output: At least one quantity is produced.
- Definiteness: Each instruction is clear and unambiguous.
- Finiteness: If we trace out the instructions of an algorithm then for all cases, the algorithm terminates after a finite number of steps.
- Effectiveness: Every instruction must be very basic so that it can be carried out, in principle, by a person using only pencil and paper. It is not enough that each operation be definite as in criterion 3, it also must be feasible. → **6Marks**

Any example algorithm → **4 Marks**

2. List and define various basic efficiency classes. [10 Marks]

Class	Name	Comments
1	*constant*	Short of best-case efficiencies, very few reasonable examples can be given since an algorithm's running time typically goes to infinity when its input size grows infinitely large.
$\log n$	*logarithmic*	Typically, a result of cutting a problem's size by a constant factor on each iteration of the algorithm
$n \log n$	"*n-log-n*"	Many divide-and-conquer algorithms fall into this category.
n^2	*quadratic*	Typically, characterizes efficiency of algorithms with two embedded loops. Elementary sorting algorithms and certain operations on *n*-by-*n* matrices are standard examples.
n^3	*cubic*	Typically, characterizes efficiency of algorithms with three embedded loops
2^n	*exponential*	Typical for algorithms that generate all subsets of an *n*-element set. Often, the term "exponential" is used in a broader sense to include this and larger orders of growth as well.
$n!$	*factorial*	Typical for algorithms that generate all permutations of an *n*-element set.

3. Explain asymptotic notations Big O, Big Ω and Big θ that are used to compare the order of growth of an algorithm with example. [3+3+4= 10 Marks]

Big O : A function t(n) is said to be in O(g(n)), denoted $t(n) \in O(g(n))$, if t(n) is bounded above by some constant multiple of g(n) for all large n, i.e., if there exist some positive constant c and some nonnegative integer n0 such that

$t(n) \leq cg(n)$ for all $n \geq n0$ → **1 Mark**

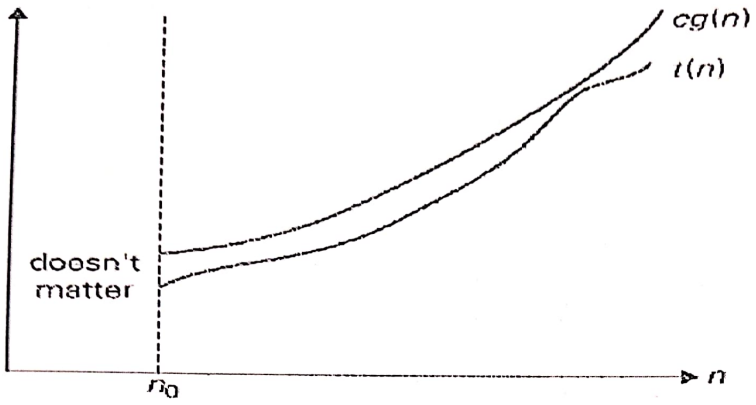

FIGURE 2.1 Big-oh notation: $t(n) \in O(g(n))$

→ **1 Mark**

Any example: ½ $n^2 \leq n^2$ for all $n \geq 0$, → **1 Mark**

Big Ω: A function *t(n)* is said to be in *Ω(g(n))*. denoted $t(n) \in \Omega(g(n))$, if *t(n)* is bounded below by some positive constant multiple of *g(n)* for all large *n,* i.e., if there exist some positive constant *c* and some nonnegative integer *n0* such that

$t(n) \geq cg(n)$ for all $n \geq no$.

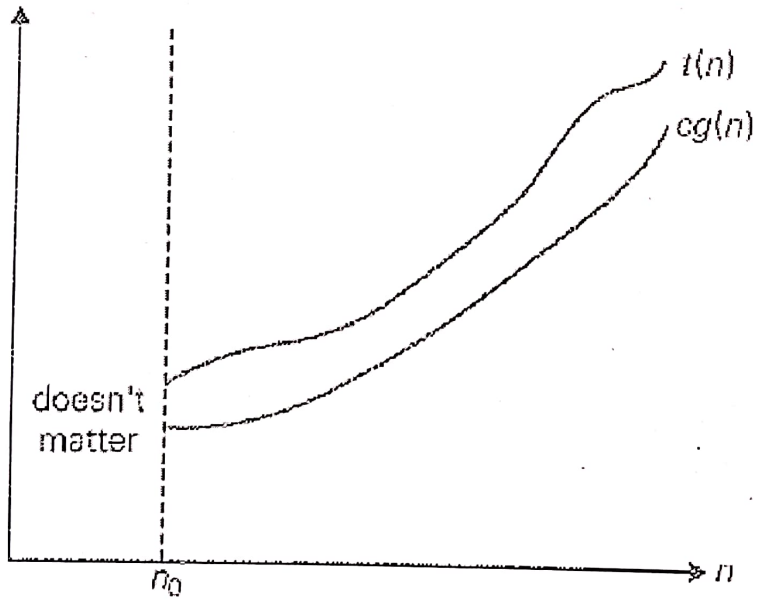

FIGURE 2.2 Big-omega notation: $t(n) \in \Omega(g(n))$

→ **1 Mark**

Any Example: $n^3 \geq n^2$ for all $n \geq 0$, → **1 Mark**

Big Θ: A function *t(n)* is said to be in *Θ(g(n))*, denoted $t(n) \in \Theta(g(n))$, if *t(n)* is bounded both above and below by some positive constant multiples of *g(n)* for all large *n,* i.e., if there exist some positive constant c1 and c2 and some nonnegative integer *n0* such that

- $c2g(n) \leq t(n) \leq c1g(n)$ for all $n \geq n0$. → **2 Marks**

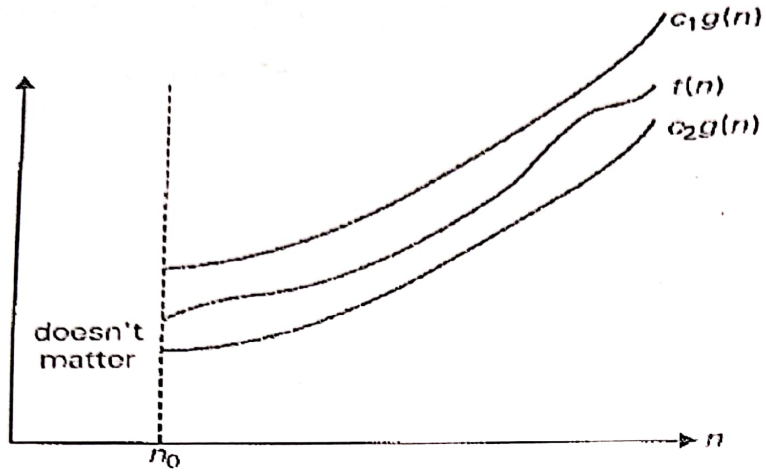

FIGURE 2.3 Big-theta notation: $t(n) \in \Theta(g(n))$ → **1 Mark**

Any Example: ½ n(n-1) $\epsilon\ \Theta(n^2)$ for all $n \geq 2$, → **1 Mark**

4. Explain general plan of mathematical analysis of non-recursive Algorithms with example. [6+4=10 Marks]

1. Decide on a parameter (or parameters) indicating an input's size.
2. Identify the algorithm's basic operation. (As a rule, it is located in its innermost loop.)
3. Check whether the number of times the basic operation is executed depends only on the size of an input. If it also depends on some additional property, the worst-case, average-case, and, if necessary, best-case efficiencies have to be investigated separately.
4. Set up a sum expressing the number of times the algorithm's basic operation is executed.
5. Using standard formulas and rules of sum manipulation either find a closed form formula for the count or, at the very least, establish its order of growth. →**6 Marks**

Any example algorithm with efficiency → **4 marks**

5. Define best-case, worst-case and average-case efficiency. Write the algorithm and give these efficiencies for sequential search. [3+3+4=10marks] OR

- **The *worst-case efficiency*** *of an algorithm is its efficiency for the worst-case* input of size *n, which is an input (or inputs) of size n for which the algorithm* runs the longest among all possible inputs of that size.
- The ***best-case efficiency*** *of an algorithm is its efficiency for the best-case input* of size *n, which is an input (or inputs) of size n for which the algorithm runs the* fastest among all possible inputs of that size.

- Neither the worst-case analysis nor its best –case yields the necessary information about an algorithm's behavior on a "typical" or "random" input. This is the information that the ***average-case efficiency*** *seeks to provide.* → **3 Marks**

ALGORITHM *SequentialSearch(A[O .. n - 1], K)*

//Searches for a given value in a given array by sequential search

//Input: An array *A[O .. n -1] and a search key K*

//Output: The index of the frrst element of *A that matches K or -1 if there are no //matching elements*

i← 0

while i <n and A[i] ≠ *K do*

i ← i+1

if i < n return i

else return -1 → **3 Marks**

$C_{worst}(n) = n.$ → **1 Marks**

$C_{best}(n) = 1.$ → **1 Marks**

$$C_{avg}(n) = [1 \cdot \frac{p}{n} + 2 \cdot \frac{p}{n} + \cdots + i \cdot \frac{p}{n} + \cdots + n \cdot \frac{p}{n}] + n \cdot (1 - p)$$

$$= \frac{p}{n}[1 + 2 + \cdots + i + \cdots + n] + n(1 - p)$$

$$= \frac{p}{n}\frac{n(n+1)}{2} + n(1 - p) = \frac{p(n+1)}{2} + n(1 - p).$$

$$C_{avg}(n) = \frac{p(n+1)}{2} + n(1 - p).$$

→ **2 Marks**

6. Write the algorithm to find maximum element in the given array and explain the mathematical analysis of this non-recursive algorithm. [5+5=10marks]

ALGORITHM *MaxElernent(A[O .. n* -1])

//Determines the value of the largest element in a given array

//Input: An array *A[0 .. n* - 1] of real numbers

//Output: The value of the largest element in *A*

maxval ←*A[0]*

for i ←1 to n-1 do

if *A[i]* > *maxval*

maxval ← *A[i]*

return *maxval* **→5 Marks**

- Let us denote *C(n)* the number of times this comparison is executed.
- The algorithm makes one comparison on each execution of the loop, which is repeated for each value of the loop's variable *i* within the bounds 1 and *n* - 1 (inclusively). Therefore, we get the following sum for C(n):

$$C(n) = \sum_{i=1}^{n-1} 1 = n - 1 \in \Theta(n).$$

→5 Marks

7. Write the algorithm to check whether all the elements in the given array are distinct and explain the mathematical analysis of this non- recursive algorithm. Derive its worst-case time complexity. [5+5=10 Marks] OR

ALGORITHM *UniqueElements(A[O .. n* -1])

//Determines whether all the elements in a given array are //distinct

//Input: An array *A[O .. n* -1]

//Output: Returns "true" if all the elements in *A* are distinct

II and "false" otherwise

for *i* ← 0 to *n* - 2 do

for *j* ← i + 1 to *n* - 1 do

if *A[i]* = *A[j]* return false

return true **→ 5 Marks**

$$C_{worst}(n) = \sum_{i=0}^{n-2} \sum_{j=i+1}^{n-1} 1 = \sum_{i=0}^{n-2} [(n-1) - (i+1) + 1] = \sum_{i=0}^{n-2} (n-1-i)$$

$$= \sum_{i=0}^{n-2} (n-1) - \sum_{i=0}^{n-2} i = (n-1) \sum_{i=0}^{n-2} 1 - \frac{(n-2)(n-1)}{2} = (n-1)[(n-2)-0+1] - \frac{(n-2)(n-1)}{2}$$

$$= (n-1)^2 - \frac{(n-2)(n-1)}{2} = \frac{(n-1)n}{2} \approx \frac{1}{2}n^2 \in \Theta(n^2).$$

Note that this result was perfectly predictable: in the worst case, the algorithm needs to compare all $n(n-1)/2$ distinct pairs of its n elements.

→ 5 Marks

8. Write the algorithm to perform matrix multiplication and explain the mathematical analysis of this non-recursive algorithm[5+5=10 Marks]

ALGORITHM *MatrixMultiplication(A[O .. n- 1, O .. n- 1], B[O .. n- 1, O .. n- 1])*

//Multiplies two *n-by-n* matrices by the definition-based //algorithm

//Input: Two *n-by-n* matrices *A* and *B*

//Output: Matrix C = *AB*

for i ←*O to n* **-1 do**

for *j* ←0 to *n* - 1 **do**

C[i, j] ←0.0

for *k* ←0 **to** *n* - 1 do

C[i, j] ←*C[i,* j] + *A[i,* k] * *B[k,* j]

return C → **5 Marks**

- There is just one multiplication executed on each repetition of the algorithm's innermost loop, which is governed by the variable *k* ranging from the lower bound 0 to the upper bound *n* - 1. Therefore, the number of multiplications made for every pair of specific values of variables *i* and *j* is

$$\sum_{k=0}^{n-1} 1,$$

- The total number of multiplications *M(n)* is expressed by the following triple sum

$$M(n) = \sum_{i=0}^{n-1}\sum_{j=0}^{n-1}\sum_{k=0}^{n-1} 1. \qquad T(n) \approx c_m M(n) = c_m n^3,$$

$$= \sum_{i=0}^{n-1}\sum_{j=0}^{n-1} n \qquad T(n) \approx c_m M(n) + c_a A(n) = c_m n^3 + c_a n^3 = (c_m + c_a) n^3,$$

$$= \sum_{i=0}^{n-1} n^2$$

$$= n^3.$$

→ **5 Marks**

9. Write the recursive algorithm to find the Factorial of a given number. and illustrate its mathematical analysis. [4+6=10 Marks] OR

ALGORITHM *F(n)*

//Computes *n!* recursively

//Input: A nonnegative integer *n*

//Output: The value of *n!*

if *n* = 0 return 1

else return *F(n* - 1) * *n* → **4 Marks**

- Multiplication is the basic operation for algorithm F(n)

the number of multiplications $M(n)$ needed to compute it must satisfy the equality

$$M(n) = \underbrace{M(n-1)}_{\text{to compute } F(n-1)} + \underbrace{1}_{\text{to multiply } F(n-1) \text{ by } n} \quad \text{for } n > 0.$$

- This is called as recurrence relation.
- We need an initial condition which is used to terminate the recursion.
- It is M(0)=0 i.e., when n=0 zero number of multiplication.
- Using Backward substitution method recurrence relation can be simplified to get the solution.

M(n) = *M(n-* 1) + 1 for *n* > 0,

M(0) = 0.

M(n) = *M(n-* 1) + 1 substitute *M(n-* 1) = *M(n-* 2) + 1

= *[M(n-* 2) + 1] + 1

= *M(n-* 2) + 2 substitute *M* (n - 2) = *M (n* - 3) + 1

= *[M(n-* 3) + 1] + 2 = *M(n-* 3) + 3.

In general for the value of i

M(n) = *M(n- i)* + *i*

substitute i = *n to get the advantage of initial condition*

M(n)=M(n- n) + *n=M(0)+n=0+n=n* → **6 Marks**

10. Illustrate mathematical analysis of recursive algorithm for Towers of Hanoi. [4+6=10 Marks]

Step 1: if $n = 1$, move the single disk directly from the source peg to the destination peg.

Step 2: If n>1 move recursively $n - 1$ disks from peg 1 to peg 2 (with peg 3 as auxiliary), (n-1 moves)

Step 3: move the largest disk directly from peg 1 to peg 3, (1 move)

Step 4: move recursively $n - 1$ disks from peg 2 to peg 3 (using peg 1 as auxiliary). (n-1 move)

M(n)=M(n-1)+1+M(n-1) when n>1→ recurrence relation

M(1)=1 when n=1 → initial condition → **4 Marks**

- **Apply backward substitution method to solve the recurrence**

$M(n) = M(n-1) + 1 + M(n-1)$ for $n > 1$. $M(1) = 1$

$M(n) = 2M(n-1) + 1$ for $n > 1$

$M(n) = 2M(n-1) + 1$ sub. $M(n-1) = 2M(n-2) + 1$

$= 2[2M(n-2) + 1] + 1$

$= 2^2M(n-2) + 2 + 1$ sub. $M(n-2) = 2M(n-3) + 1$

$= 2^2[2M(n-3) + 1] + 2 + 1$

$= 2^3M(n-3) + 2^2 + 2 + 1 = 2^3M(n-3) + 2^3 - 1$

$= 2^4M(n-4) + 2^3 + 2^2 + 2 + 1 = 2^4M(n-4) + 2^4 - 1$

and, generally, after i substitutions, we get

$M(n) = 2^iM(n-i) + 2^{i-1} + 2^{i-2} + \cdots + 2 + 1 = 2^i M(n-i) + 2^i - 1.$

Put $i = n - 1$, as initial condition is for n=1

$M(n)=2^{n-1}M(n-(n-1))+2^{n-1} - 1$

$= 2^{n-1} M(1) + 2^{n-1} - 1 = 2^{n-1} + 2^{n-1} - 1 = 2.2^{n-1} - 1 = 2^{n-1+1} - 1 = 2^n - 1$ → **6 Marks**

ADICHUNCHANAGIRI INSTITUTE OF TECHNOLOGY, CHIKAMAGALURU

DEPARTMENT OF INFORMATION SCIENCE & ENGINEERING

Academic Year: 2021-22 **Laboratory Rubrics** Semester:4

Course Name and Code: Microcontroller and Embedded Systems Laboratory (18CSL48)

Phase1: Daily Lab Rubrics (16 Marks) : MS. Bharan. V (4AI20IS009).

Experiment No	Date	Pre-lab preparation (Observation, Writing) -3M	Debugging and Execution- 8M	Lab Record- 3M	Attendance – 2M	Total
1	25/5	3	8	3	A	14
2	25/5	2	8	3	2	16
3	1/6	3	8	3	2	16
4	1/6	3	8	3	2	16
5	8/6	3	8	3	2	16
6	8/6	3	8	3	2	16
7	15/6	3	8	3	2	16
✓✓ 8 ✓✓ 4/5 6/15	15/6	3	8	3	2	16
9	22/6	3	8	3	2	16
10	29/6	3	8	3	2	16
11	6/7	3	8	3	2	16
12	13/7	3	8	3	2	16
13	20/7	3	8	3	2	16
14	27/7	3	8	3	2	16
15	10/8	3	8	3	2	16
16	24/8	3	8	3	2	16
					Average Marks	/16

Phase2: Final Lab Internal Assessment Rubrics (24 Marks)

Experiment	Procedure-7M	Debugging and Execution - 12M	Viva – 5 M	Total
Part A Pgm No - 4, 5, 6, 7	7	12	5	24
Part B Pgm No - 15, 7	7	12	5	24
			Average Marks	24 /24

Phase 1 Marks (P1)	Phase 2 Marks (P2)	Total Marks(P1 +P2)
16	24	40

Anjali

Course Instructor Signature

HOD's Signature

Lab Rubrics of student (IV sem): 2021-22