



**ANIL NEERUKONDA INSTITUTE OF TECHNOLOGY & SCIENCES  
(UGC AUTONOMOUS)**

(Affiliated to AU, Approved by AICTE & Accredited by NBA & NAAC with 'A' Grade)  
Sangivalasa 531 162, Bheemunipatnam Mandal, Visakhapatnam Dist

---

**DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING**

**Date: 18-08-2022**

**CIRCULAR**

The Department of Computer Science and Engineering, ANITS is organizing an interactive session on "GATE EXAMINATION". This event is organised by Mamatha.V, Assistant Professor, CSE . It was held on 22.08.2022 to 26.08.2022 for the B.Tech I/IV , II/IV B.Tech CSE students of ANITS.

**Title: " GATE EXAMINATION"**

**Date: 22-08-2022 to 26-08-2022**

**Time: 10:00 Am -12:00Pm**

**&**

**01:00 Pm -03:00Pm**

**Venue: ECE Seminar Hall (B-101)**

**Faculty Coordinator  
Mrs.Mamatha.V**

**HOD, CSE  
(Dr. P.E.S.N.Krishna Prasad)**

Head of the Department of  
Computer Science & Engineering  
Anil Neerukonda Institute of  
Technology & Sciences  
Sangivalasa, Visakhapatnam Dist



**ANIL NEERUKONDA INSTITUTE OF TECHNOLOGY & SCIENCES**  
(UGC AUTONOMOUS)  
(Affiliated to AU, Approved by AICTE & Accredited by NBA & NAAC with 'A' Grade)  
Sangivalasa 531 162, Bheemunipatnam Mandal, Visakhapatnam Dist

**DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING**

**CAREER GUIDANCE CELL - CSE**

**2/4 CSE-A,B,C**

**Date: 18-08-2022**

S.no	Date	Topic		Speaker
1	22.08.2022	GATE Importance	Examination Pattern Syllabus	Dr.P.E.S.N.Krishna Prasad
2	23.08.2022	Database Management Systems	SQL Normalization	Mamatha V
3	24.08.2022	Compiler Design	Compiler Design	K.Amaravati
4	25.08.2022	Data Structures	Stacks,Queues Linked Lists Trees,Graphs	D.Ashwini
5	26.08.2022	Programming	C Programming, Java Programming	Dr.V.Sangeetha

*Mamatha V*  
Faculty Coordinator  
Mrs.Mamatha.V

*Dr. P.E.S.N. Krishna Prasad*  
HOD, CSE  
(Dr. P.E.S.N.Krishna Prasad) 18/8

Head of the Department of  
Computer Science & Engineering  
Anil Neerukonda Institute of  
Technology & Sciences  
Sangivalasa, Visakhapatnam Dist



**Anil Neerukonda Institute of Technology & Sciences  
(Autonomous)**

(Affiliated to AU, Approved by AICTE & Accredited by NBA (ECE, EE, CSE, IT & Mech.) & NAAC)  
Sangivalasa- 533 162, Bheemuniapatnam Mandal, Visakhapatnam District  
Phone: 08933-225083/4/87 Fax: 226395  
Website: www.anits.edu.in email: principal@anits.edu.in

Date:22-08-2022

GATE  
Topic: Gate Importance  
Faculty-Dr. P.E.S.N Krishna Prasad, Hod

On 22-08-2022, Dr.Krishna Prasad addressed the students on the Gate Examination pattern and syllabus to all the students. It was an interactive session with the students where the students understood all the topics clearly.

**Abstract:**

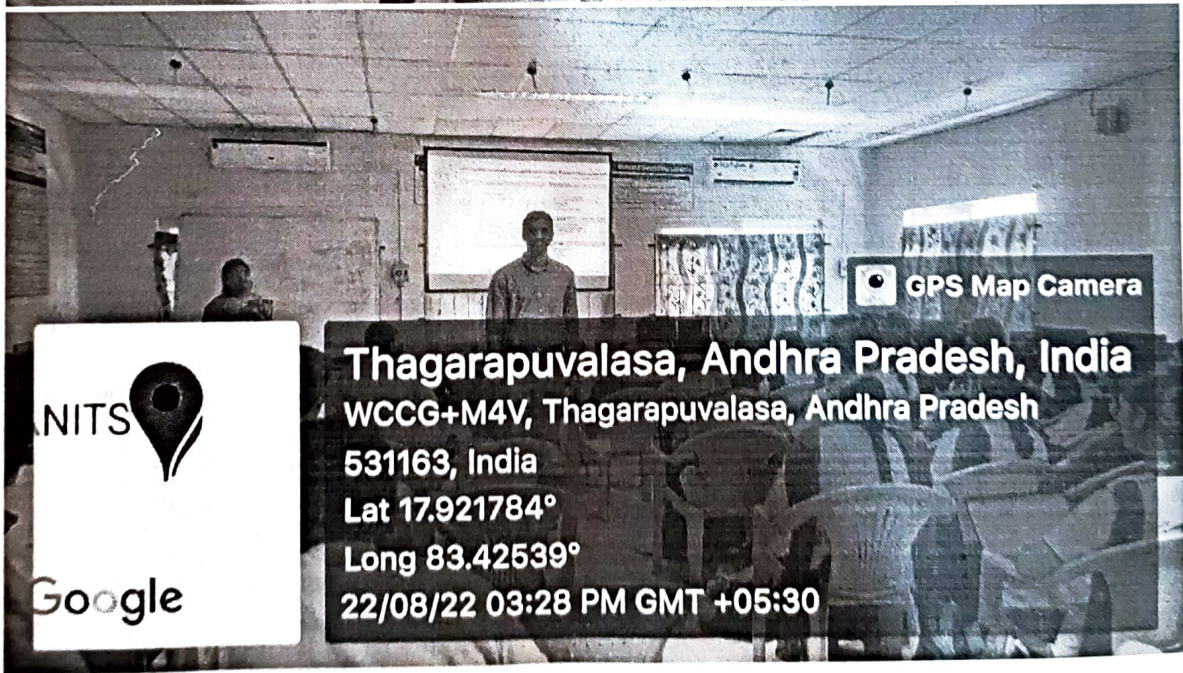
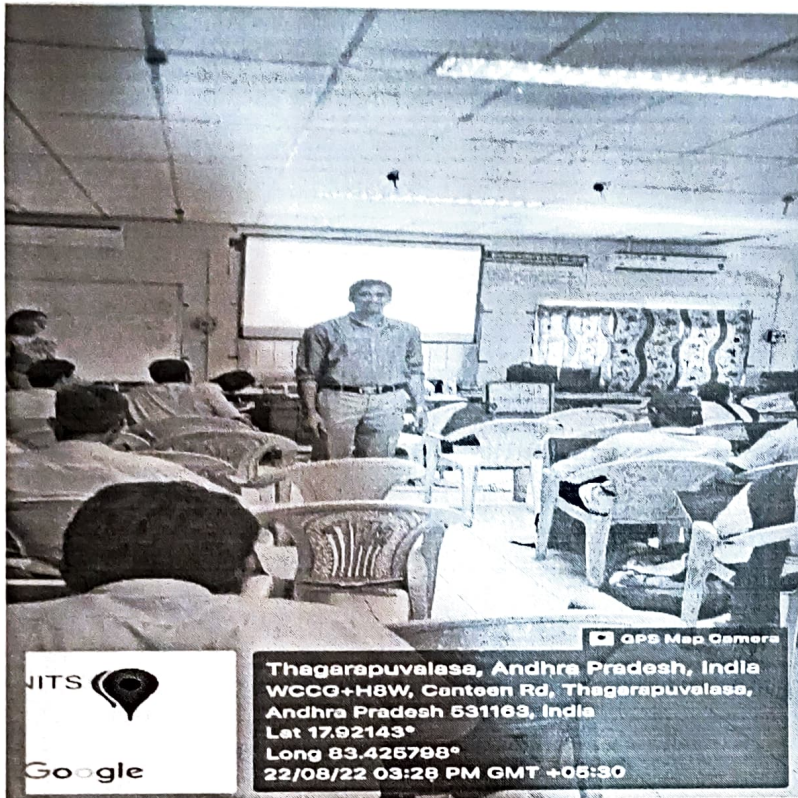
The Graduate Aptitude Test in Engineering (GATE) is an examination conducted in India that primarily tests the comprehensive understanding of various undergraduate subjects in engineering and science for admission into the Masters's Program and Job in Public Sector Companies. GATE is conducted jointly by the Indian Institute of Science and seven Indian Institutes of Technologies at Roorkee, Delhi, Guwahati, Kanpur, Kharagpur, Chennai (Madras), and Mumbai (Bombay) on behalf of the National Coordination Board – GATE, Department of Higher Education, Ministry of Education (MoE), Government of India.

A candidate's GATE score reflects the candidate's relative performance level. The score is used for admissions to various post-graduate education programs (e.g. Master of Engineering, Master of Technology, Master of Architecture, Doctor of Philosophy) in Indian higher education institutes, with financial assistance provided by MHRD and other government agencies. Recently, GATE scores are also being used by several Indian public sector undertakings for recruiting graduate engineers in entry-level positions. It is one of the most competitive examinations in India. GATE is also recognized by various institutes outside India, such as Nanyang Technological University in Singapore.

## GATE Exam Pattern

<b>Particulars</b>	<b>Details</b>
<b>Full Exam Name</b>	Graduate Aptitude Test in Engineering
<b>Short Exam Name</b>	GATE
<b>Managing By</b>	Indian Institute of Technology Kharagpur
<b>Exam Level</b>	National
<b>Languages</b>	English
<b>Mode of Application</b>	Online
<b>Application Fee (General)</b>	1500
<b>Mode of Exam</b>	Online
<b>Mode of Question Papers</b>	MSQ/MCQ/NAT
<b>Participating Colleges</b>	1072
<b>Exam Duration</b>	3 Hours

Photos:



Attendance:

*S. S. J. S.*  
Head of the Department of  
Computer Science & Engineering  
Anil Neerakonda Institute of  
Technology & Sciences  
Sangivalasa, Visakhapatnam, India

22/12/22

ANIL NEERUKONDA INSTITUTE OF TECHNOLOGY AND SCIENCES (A)  
DEPARTMENT OF CSE  
3-4 B. Tech CSE - A SEM - I

GATE Classes

S.No	Roll No	Induction Programme Name of the student	Signature
1	320126510001	ADAPA SASHANK	[Signature]
2	320126510002	ADAPAKA NEELIMA	[Signature]
3	320126510003	ADINHOTRAM SRI VENKATA SAIKIRAN CHAITANYA	[Signature]
4	320126510004	ANNEPU HEMANTH KUMAR	[Signature]
5	320126510005	BANDARU VENKATA JAYARAM	[Signature]
6	320126510006	BETHA AJAY KUMAR	[Signature]
7	320126510007	BETHA AKASH	[Signature]
8	320126510008	BITRA VENKAT SAI KIRAN	[Signature]
9	320126510009	BODDU THANMAI	[Signature]
10	320126510010	BOOSI ARAVIND	[Signature]
11	320126510011	BUDHA GOWTHAM	[Signature]
12	320126510012	BULASARA LIKHIT SRI SAI	[Signature]
13	320126510013	CHIGATI SATWIKA	[Signature]
14	320126510014	CHINNAPARAPU AJAY ESWAR	[Signature]
15	320126510015	CHINTA DEEPTHI	[Signature]
16	320126510016	DEVIKA KONDAPALLI	[Signature]
17	320126510017	G SUNIL	[Signature]
18	320126510018	GANDI SARAN KUMAR	[Signature]
19	320126510019	GANGADEVI LAKSHMI PRAVALLIKA	[Signature]
20	320126510020	GELLA CHAITANYA VENKATA SAI	[Signature]
21	320126510021	GOPSETTI ANIL KUMAR	[Signature]
22	320126510022	GOSALA SURYA YARDHAN REDDY	[Signature]
23	320126510023	GUDIWADA NAGA LAKSHMI NAVEEN	[Signature]
24	320126510024	GUDLA TEJESH	[Signature]
25	320126510025	GUNDU ABHINAYA	[Signature]
26	320126510026	INTI AAKASH	[Signature]
27	320126510027	JARAJANA HARSHIT	[Signature]
28	320126510028	KARICHETI SIVA KALYAN	[Signature]
29	320126510029	KASAMSETTI MANOJ SAMPATH	[Signature]
30	320126510030	KASINA CHANDI NAGA PAVAN RAJESH	[Signature]
31	320126510031	KASIREDDI SAI SHRUTHI	[Signature]
32	320126510032	KOLLI ANANTHA RAO	[Signature]
33	320126510033	KOMPILLI RAJ KUMAR	[Signature]
34	320126510034	KONDAVETI ANUPAMA	[Signature]
35	320126510035	KOTA CHANDINI MANOZNA	[Signature]
36	320126510036	KUMBHA NARENDRA KUMAR	[Signature]
37	320126510037	KURAKULA ROHIT	[Signature]
38	320126510038	LALITH KIRAN PEERUKATTLA	[Signature]
39	320126510039	LODAGALA HEMA SAI	[Signature]
40	320126510040	M MOHIDEEN FATHIMA NILOFAR	[Signature]
41	320126510041	MAHATO USHA	[Signature]
42	320126510042	MARADA JOSHIITH	[Signature]
43	320126510043	MATTA BHASHITHA	[Signature]
44	320126510044	METTU AKHILESH	[Signature]

[Handwritten mark]



Head of the Department of  
Computer Science & Engineering  
Anil Neerukonda Institute of  
Technology and Sciences  
[Signature]

S.No	Roll No	Name of the student	Signature
45	320126510043	MOLLETI YERNI TRIBHUVAN	<i>[Signature]</i>
46	320126510046	MUMMIDIVARAPU VENKATESH	<i>[Signature]</i>
47	320126510048	NAMMI ADITYA	<i>[Signature]</i>
48	320126510049	PALLE TARUN KUMAR	<i>[Signature]</i>
49	320126510050	PANANGIPALLI KARTHIKEYA	<i>[Signature]</i>
50	320126510051	PANDA ANJALI	<i>[Signature]</i>
51	320126510052	PASUPUREDDI HARITHA	<i>[Signature]</i>
52	320126510053	PATTAMATTA NAGA HIMA VARSHA	<i>[Signature]</i>
53	320126510054	PATTIKA RAJESH	<i>[Signature]</i>
54	320126510055	PUNUGOTI CHANDRA SEKJAR	<i>[Signature]</i>
55	320126510056	RAGHAVENDRA ANURAG SETAMRAJU	<i>[Signature]</i>
56	320126510057	RUSHIKESH MUKUND PATMASE	<i>[Signature]</i>
57	320126510058	SALA VENKATA NAGA SAI CHARAN	<i>[Signature]</i>
58	320126510059	SANAPALA NIKHILESH	<i>[Signature]</i>
59	320126510060	SEEPANA MITHUN	<i>[Signature]</i>
60	320126510061	SHAIK BASHIEER AHAMED	<i>[Signature]</i>
61	320126510062	VANAM LALITHA SHREYA	<i>[Signature]</i>
62	320126510063	VEMA VENKATA YESWANTH GUPTHA	<i>[Signature]</i>
63	320126510064	VYSYARAJU RAMAN RAJ	<i>[Signature]</i>
64	321126510L01	KARRI SAI KUMAR	<i>[Signature]</i>
65	321126510L02	MAJJI HEMANTH REDDY	<i>[Signature]</i>
66	321126510L03	KURMAPU DHARANEESWAR	<i>[Signature]</i>
67	321126510L04	DONI TWINKLE	<i>[Signature]</i>
68	321126510L05	AMARA NAGENDRA	<i>[Signature]</i>
69	321126510L06	SABBAVARAPU USHA SRI	<i>[Signature]</i>

*[Handwritten mark]*



*[Handwritten Signature]*  
 Head of the Department of  
 Computer Science & Engineering  
 Anil Neerukonda Institute of  
 Technology & Sciences  
 Sangivalasa, Visakhapatnam Dist

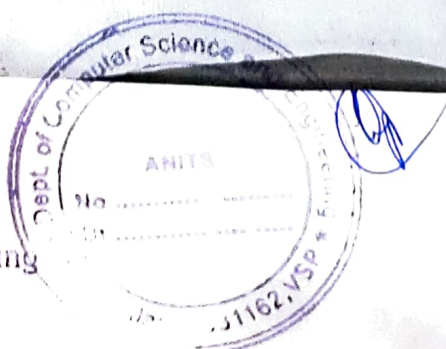
22/8/22

ANIL NEERUKONDA INSTITUTE OF TECHNOLOGY AND SCIENCES (A)  
DEPARTMENT OF CSE  
3-4 B. Tech CSE - A SEM - I

GATE Class

S.No	Roll No	Name of the student	Signature
1	320126510001	ADAPA SASHANK	Sashank
2	320126510002	ADAPAKA NEELIMA	Neelima
3	320126510003	AGNIBHTRAM SRI VENKATA SAIKIRAN CHAITANYA	Chaitanya
4	320126510004	ANNENU HEMANTH KUMAR	
5	320126510005	BANDARU VENKATA JAYARAM	
6	320126510006	BETHA AJAY KUMAR	Ajay Kumar
7	320126510007	BETHA AKASH	Akash
8	320126510008	BITRA VENKAT SAI KIRAN	Bitra
9	320126510009	BODDU THANMAI	Thanmai
10	320126510010	BOOSI ARAVIND	Aravind
11	320126510011	BUDHA GOWTHAM	Gowtham
12	320126510012	BULASARA LIKHIT SRI SAI	Likhith
13	320126510013	CHIGATI SATWIK	Satwik
14	320126510014	CHINNAPARAPU AJAY ESWAR	Ajay Eswar
15	320126510015	CHINTA DEEPTHI	Deepti
16	320126510016	DEVIKA KONDAPALLI	Devika
17	320126510017	G SUNIL	Sunil
18	320126510018	GANDI SARAN KUMAR	Saran
19	320126510019	GANGADEVI LAKSHMI PRAVALLIKA	Pravallika
20	320126510020	GELLA CHAITANYA VENKATA SAI	Chaitanya
21	320126510021	GOPSETTI ANIL KUMAR	Anil Kumar
22	320126510022	GOSALA SURYA VARDHAN REDDY	Surya
23	320126510023	GUDIWADA NAGA LAKSHMI NAVEEN	Naveen
24	320126510024	GUDLA TEJESH	Tejesh
25	320126510025	GUNDU ABHINAYA	Abhinaya
26	320126510026	INTI AAKASHI	Aakashi
27	320126510027	JARAJANA HARSHIT	Harshit
28	320126510028	KARICHIETI SIVA KALYAN	Siva
29	320126510029	KASAMSETTI MANOJ SAMPATH	Manoj
30	320126510030	KASINA CHANDI NAGA PAVAN RAJESH	Pavan
31	320126510031	KASIREDDI SAI SHRUTHI	Shruthi
32	320126510032	KOLLI ANANTHA RAO	Anantha
33	320126510033	KOMPILLI RAJ KUMAR	Raj Kumar
34	320126510034	KONDAVETI ANUPAMA	Anupama
35	320126510035	KOTA CHANDINI MANOZNA	Chandini
36	320126510036	KUMBHA NARENDRA KUMAR	Narendra
37	320126510037	KURAKULA ROHIT	Rohit
38	320126510038	LALITH KIRAN PEERUKATLA	Lalith
39	320126510039	LODAGALA HEMA SAI	Hema
40	320126510040	M MOHIDEEN FATHIMA NILOFAR	Fathima
41	320126510041	MAHATO USHA	Usha
42	320126510042	MARADA JOSHI	Joshi
43	320126510043	MATTA BHASHITHA	Bhashitha
44	320126510044	MIETTU AKHILESH	Akhilesh

Head of the Department of  
Computer Science & Engineering  
Anil Neerukonda Institute of  
Technology & Sciences  
Sangivalasa, Visakhapatnam Dist





S.No	Roll No	Name of the student	Signature
45	320126510045	MOLLEJI YERNI TRIBHUVAN	<i>Molleji</i>
46	320126510046	MUMMIDIVARAPU VENKATESH	<i>Mummidi</i>
47	320126510048	NAMMI ADITYA	<i>Nammi</i>
48	320126510049	PALLE TARUN KUMAR	<i>Tarun</i>
49	320126510050	PANANGIPALLI KARTHIKEYA	<i>Karthik</i>
50	320126510051	PANDA ANJALI	<i>Anjali</i>
51	320126510052	PASUPUREDDI HARITHA	<i>Haritha</i>
52	320126510053	PATTAMATTA NAGA HIMA VARSHA	<i>Varsha</i>
53	320126510054	PAATIKA RAJESH	<i>Rajesh</i>
54	320126510055	PUNUGOTI CHANDRA SEKHAR	<i>Chandra</i>
55	320126510056	RAGHAVENDRA ANURAG SETAMRAJU	<i>Anurag</i>
56	320126510057	RUSHIKESH MUKUND PATMASE	<i>Mukund</i>
57	320126510058	SALA VENKATA NAGA SAI CHARAN	<i>Sai Charan</i>
58	320126510059	SANAPALA NIKHILESH	<i>Nikhil</i>
59	320126510060	SEEPANA MITHUN	<i>Mithun</i>
60	320126510061	SHAIK BASHEER AHAMED	<i>Basheer</i>
61	320126510062	VANAM LALITHA SHREYA	<i>Shreya</i>
62	320126510063	VEMA VENKATA YESWANTH GUPTHA	<i>Yeswanth</i>
63	320126510064	VYSYARAJU RAMAN RAJ	<i>Raman Raj</i>
64	321126510L01	KARRI SAI KUMAR	<i>Sai Kumar</i>
65	321126510L02	MAJJI HEMANTH REDDY	<i>Hemant</i>
66	321126510L03	KURMAPU DHARANEESWAR	<i>Dharaneeswar</i>
67	321126510L04	DONI TWINKLE	<i>Twinkle</i>
68	321126510L05	AMARA NAGENDRA	<i>Nagendra</i>
69	321126510L06	SABBAVARAPU USHA SRI	<i>Usha Sri</i>



*[Signature]*  
Head of the Department of  
Computer Science & Engineering  
Anil Neerukonda Institute of  
Technology & Sciences  
Sangivalasa, Visakhapatnam

22/8/22

## ANIL NEERUKONDA INSTITUTE OF TECHNOLOGY AND SCIENCES (A)

DEPARTMENT OF CSE

GATE Classes


## 3-4 B. Tech CSE - A SEM - I

## Induction Programme

S.No	Roll No	Name of the student	Signature
1	320126510001	ADAPA SASHANK	Sashank
2	320126510002	ADAPAKA NEELIMA	Neelima
3	320126510003	AGNIHOTRAM SRI VENKATA SAIKIRAN CHAITANYA	Sai Kiran Chaitanya
4	320126510004	ANNEPU HEMANTH KUMAR	Hemant
5	320126510005	BANDARU VENKATA JAYARAM	Jayaram
6	320126510006	BETHA AJAY KUMAR	AJK
7	320126510007	BETHA AKASH	Akash
8	320126510008	BITRA VENKAT SAI KIRAN	B De Sairathi
9	320126510009	BODDU THANMAI	Thanmai
10	320126510010	BOOSI ARAVIND	Aravind
11	320126510011	BUDHA GOWTHAM	Gowtham
12	320126510012	BULASARA LIKHIT SRI SAI	Likhith Sri Sai
13	320126510013	CHIGATI SATWIKA	Satwika
14	320126510014	CHINNAPARAPU AJAY ESWAR	AJESwar
15	320126510015	CHINTA DEEPTHI	Deepthi
16	320126510016	DEVIKA KONDAPALLI	Devika
17	320126510017	G SUNIL	Sunil
18	320126510018	GANDI SARAN KUMAR	Saran Kumar
19	320126510019	GANGADEVI LAKSHMI PRAVALLIKA	
20	320126510020	GELLA CHAITANYA VENKATA SAI	Chaitanya
21	320126510021	GOPISETTI ANIL KUMAR	Anil
22	320126510022	GOSALA SURYA VARDHAN REDDY	SURYA
23	320126510023	GUDIWADA NAGA LAKSHMI NAVEEN	
24	320126510024	GUDLA TEJESH	Tejesh
25	320126510025	GUNDU ABHINAYA	Abhi
26	320126510026	INTI AAKASH	Aakash
27	320126510027	JARAJANA HARSHIT	
28	320126510028	KARICHETI SIVA KALYAN	Sivakalyan
29	320126510029	KASAMSETTI MANOJ SAMPATH	Manoj
30	320126510030	KASINA CHANDI NAGA PAVAN RAJESH	Naveen
31	320126510031	KASIREDDI SAI SHRUTHI	Sai Shruthi
32	320126510032	KOLLI ANANTHA RAO	Ananth
33	320126510033	KOMPILLI RAJ KUMAR	Raj
34	320126510034	KONDAVETI ANUPAMA	Anupama
35	320126510035	KOTA CHANDINI MANOZNA	Manoyna
36	320126510036	KUMBHA NARENDRA KUMAR	Narendra
37	320126510037	KURAKULA ROHIT	Rohit
38	320126510038	LALITH KIRAN PEERUKATTLA	Lalith
39	320126510039	LODAGALA HEMA SAI	Hema Sai
40	320126510040	M MOHIDEEN FATHIMA NILOFAR	Nilofar
41	320126510041	MAHATO USHA	Usha
42	320126510042	MARADA JOSHIITH	Joshiith
43	320126510043	MATTA BHASHIITHA	Bhashiitha
44	320126510044	METTU AKHILESH	Akhilesh

A

S.No	Roll No	Name of the student	Signature
45	320126510045	MOLLETI YERNI TRIBHUVAN	Tribhuvan
46	320126510046	MUMMIDIVARAPU VENKATESH	Venkatesh
47	320126510048	NAMMI ADITYA	N. Aditya
48	320126510049	PALLE TARUN KUMAR	Palle Tarun Kumar
49	320126510050	PANANGIPALLI KARTHIKEYA	Karthikeya
50	320126510051	PANDA ANJALI	Anjali
51	320126510052	PASUPUREDDI HARITHA	Haritha
52	320126510053	PATTAMATTA NAGA HIMA VARSHA	Varsha
53	320126510054	PATTIKA RAJESH	Rajesh
54	320126510055	PUNUGOTI CHANDRA SEKHAR	Chandhar
55	320126510056	RAGHAVENDRA ANURAG SETAMRAJU	.
56	320126510057	RUSHIKESH MUKUND PATMASE	Rishikesh
57	320126510058	SALA VENKATA NAGA SAI CHARAN	Charan
58	320126510059	SANAPALA NIKHILESH	Nikhil
59	320126510060	SEEPANA MITHUN	Mithun
60	320126510061	SHAIK BASHEER AHAMED	Basheer
61	320126510062	VANAM LALITHA SHREYA	Shreya
62	320126510063	VEMA VENKATA YESWANTH GUPTHA	Yeswanth
63	320126510064	VYSYARAJU RAMAN RAJ	Raman Raj
64	321126510L01	KARRI SAI KUMAR	
65	321126510L02	MAJJI HEMANTH REDDY	Hemant Reddy
66	321126510L03	KURMAPU DHARANEESWAR	Dharan
67	321126510L04	DONI TWINKLE	Twinkle
68	321126510L05	AMARA NAGENDRA	Nagender
69	321126510L06	SABBAVARAPU USHA SRI	Usha Sri

  
 Head of the Department of  
 Computer Science & Engineering  
 Anil Neerukonda Institute of  
 Technology & Sciences  
 Sangivalasa, Visakhapatnam Dist



22/8/22

## ANIL NEERUKONDA INSTITUTE OF TECHNOLOGY AND SCIENCES (A)

## DEPARTMENT OF CSE

GATE Classes

## 2-4 B. Tech CSE - A SEM - I

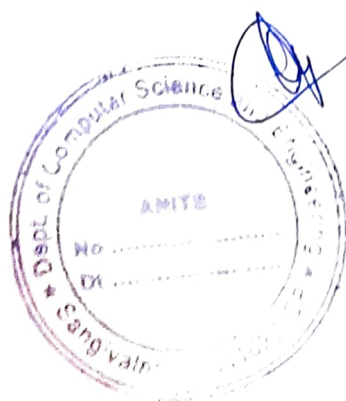
## Induction Programme

S.No	Roll No	Name of the student	Signature
1	320126510001	ADAPA SASHANK	Sashank
2	320126510002	ADAPAKA NEELIMA	Neelima
3	320126510003	AGNIHOTRAM SRI VENKATA SAIKIRAN CHAITANYA	Chaitanya
4	320126510004	ANNEPU HEMANTH KUMAR	
5	320126510005	BANDARU VENKATA JAYARAM	
6	320126510006	BETHA AJAY KUMAR	Ajay Kumar
7	320126510007	BETHA AKASH	Betha Akash
8	320126510008	BITRA VENKAT SAI KIRAN	Bitra Venkat Sai Kiran
9	320126510009	BODDU THANMAI	B. Thanmai
10	320126510010	BOOSI ARAVIND	B. Aravind
11	320126510011	BUDHA GOWTHAM	B. Gowtham
12	320126510012	BULASARA LIKHIT SRI SAI	Likhitha
13	320126510013	CHIGATI SATWIKA	Satwika
14	320126510014	CHINNAPARAPU AJAY ESWAR	Ajay Eswar
15	320126510015	CHINTA DEEPTHI	Chinta Deepthi
16	320126510016	DEVIKA KONDAPALLI	Devika
17	320126510017	G SUNIL	G. Sunil
18	320126510018	GANDI SARAN KUMAR	Saran Kumar
19	320126510019	GANGADEVI LAKSHMI PRAVALLIKA	
20	320126510020	GELLA CHAITANYA VENKATA SAI	G. Chaitanya
21	320126510021	GOPISETTI ANIL KUMAR	Anil Kumar
22	320126510022	GOSALA SURYA VARDHAN REDDY	Surya Vardhan
23	320126510023	GUDIWADA NAGA LAKSHMI NAVEEN	Naveen
24	320126510024	GUDLA TEJESH	Tejesh G.
25	320126510025	GUNDU ABHINAYA	Abhinaya
26	320126510026	INTI AAKASH	Aakash I.
27	320126510027	JARAJANA HARSHIT	
28	320126510028	KARICHETI SIVA KALYAN	Kalyan
29	320126510029	KASAMSETTI MANOJ SAMPATH	Manoj
30	320126510030	KASINA CHANDI NAGA PAVAN RAJESH	Pavan Rajesh
31	320126510031	KASIREDDI SAI SHRUTHI	Sai Shruthi
32	320126510032	KOLLI ANANTHA RAO	Anantha Rao
33	320126510033	KOMPILLI RAJ KUMAR	Raj Kumar
34	320126510034	KONDAVETI ANUPAMA	Anupama
35	320126510035	KOTA CHANDINI MANOZNA	Chandini
36	320126510036	KUMBHA NARENDRA KUMAR	Narendra
37	320126510037	KURAKULA ROHIT	Rohit
38	320126510038	LALITH KIRAN PEERUKATTLA	Kiran
39	320126510039	LODAGALA HEMA SAI	Hema Sai
40	320126510040	M MOHIDEEN FATHIMA NILOFAR	Fathima
41	320126510041	MAHATO USHA	USHA
42	320126510042	MARADA JOSHITH	Joshith
43	320126510043	MATTA BHASHITHA	Bhashitha
44	320126510044	METTU AKHILESH	

S.No	Roll No	Name of the student	Signature
45	320126510045	MOLLETI YERNI TRIBHUVAN	<i>Yerni</i>
46	320126510046	MUMMIDIVARAPU VENKATESH	<i>Venkatesh</i>
47	320126510048	NAMMI ADITYA	<i>Aditya</i>
48	320126510049	PALLE TARUN KUMAR	<i>Tarun</i>
49	320126510050	PANANGIPALLI KARTHIKEYA	<i>Karthik</i>
50	320126510051	PANDA ANJALI	<i>Anjali</i>
51	320126510052	PASUPUREDDI HARITHA	<i>Haritha</i>
52	320126510053	PATTAMATTA NAGA HIMA VARSHA	<i>Varsha</i>
53	320126510054	PATTIKA RAJESH	<i>Rajesh</i>
54	320126510055	PUNUGOTI CHANDRA SEKHAR	<i>P.C. Sekhar</i>
55	320126510056	RAGHAVENDRA ANURAG SETAMRAJU	
56	320126510057	RUSHIKESH MUKUND PATMASE	<i>Rushikesh</i>
57	320126510058	SALA VENKATA NAGA SAI CHARAN	<i>Sai Charan</i>
58	320126510059	SANAPALA NIKHILESH	<i>Nikhilesh</i>
59	320126510060	SEEPANA MITHUN	<i>Mithun</i>
60	320126510061	SHAIK BASHEER AHAMED	<i>Basheer Ahamed</i>
61	320126510062	VANAM LALITHA SHREYA	<i>Shreya V</i>
62	320126510063	VEMA VENKATA YESWANTH GUPTHA	<i>Yeswanth</i>
63	320126510064	VYSYARAJU RAMAN RAJ	<i>Raman Raj</i>
64	321126510L01	KARRI SAI KUMAR	
65	321126510L02	MAJJI HEMANTH REDDY	<i>Hemant</i>
66	321126510L03	KURMAPU DHARANEESWAR	<i>Dharaneeswar</i>
67	321126510L04	DONI TWINKLE	<i>Twinkle</i>
68	321126510L05	AMARA NAGENDRA	<i>Nagendra</i>
69	321126510L06	SABBAVARAPU USHA SRI	<i>Usha Sri</i>

*Sub. Jyoti*

Head of the Department of  
Computer Science & Engineering  
Anil Neerukonda Institute of  
Technology & Sciences  
Sangivalasa, Visakhapatnam Dist





**Anil Neerukonda Institute of Technology & Sciences**  
(Autonomous)

(Affiliated to AU, Approved by AICTE & Accredited by NBA (ECE, EEE, CSE, IT & Mech) & NAAC)  
Sangivalasa- 531 162, Bheemunipatnam Mandal, Visakhapatnam District  
Phone: 08933-22508 1/8487 Fax: 226395  
Website: www.anits.edu.in email: principal@anits.edu.in

Date:23-08-2022

Topic: Database Management Systems  
Faculty-V.Mamatha, Asst Professor

On 23-08-2022, V.Mamatha addressed the students on Database Management Systems related to Gate Examinations to all the students. It was an interactive session with the students where the students understood all the topics clearly.

**Abstract:**

- Relational Algebra
- SQL Queries
- Transactions.
- Serializability.
- Normalizations.

Relational Algebra

Relational algebra is a procedural query language. It gives a step by step process to obtain the result of the query. It uses operators to perform queries.

Types of Relational operation

DBMS Relational Algebra

1. Select Operation:

The select operation selects tuples that satisfy a given predicate.

It is denoted by sigma ( $\sigma$ ).

Notation:  $\sigma p(r)$

Where:

$\sigma$  is used for selection prediction

$r$  is used for relation

$p$  is used as a propositional logic formula which may use connectors like: AND OR and NOT.

These relational can use as relational operators like =,  $\neq$ ,  $\geq$ ,  $<$ ,  $>$ ,  $\leq$ .

2. Project Operation:

This operation shows the list of those attributes that we wish to appear in the result. Rest of the attributes are eliminated from the table.

It is denoted by  $\Pi$ .

Notation:  $\{A_1, A_2, \dots, A_n\}$  (r)

Where

$A_1, A_2, A_3$  is used as an attribute name of relation r.

### 3. Union Operation:

Suppose there are two tuples R and S. The union operation contains all the tuples that are either in R or S or both in R & S.

It eliminates the duplicate tuples. It is denoted by  $\cup$ .

Notation:  $R \cup S$

A union operation must hold the following condition:

R and S must have the attribute of the same number.

Duplicate tuples are eliminated automatically.

### 4. Set Intersection:

Suppose there are two tuples R and S. The set intersection operation contains all tuples that are in both R & S.

It is denoted by intersection  $\cap$ .

Notation:  $R \cap S$

### 5. Set Difference:

Suppose there are two tuples R and S. The set intersection operation contains all tuples that are in R but not in S.

It is denoted by intersection minus (-).

Notation:  $R - S$

### 6. Cartesian product

The Cartesian product is used to combine each row in one table with each row in the other table. It is also known as a cross product.

It is denoted by  $\times$ .

Notation:  $E \times D$

### 7. Rename Operation:

The rename operation is used to rename the output relation. It is denoted by rho ( $\rho$ ).

In this article, we will discuss the overview of SQL and will mainly focus on Concepts and Queries and will understand each with the help of examples. Let's discuss it one by one.

Overview :

SQL is a computer language that is used for storing, manipulating, and retrieving data in a structured format. This language was invented by IBM. Here SQL stands for Structured Query Language. Interacting databases with SQL queries, we can handle a large amount of data. There are several SQL-supported database servers such as MySQL, PostgreSQL, sqlite3 and so on. Data can be stored in a secured and structured format through these database servers. SQL queries are often used for data manipulation and business insights better.

SQL Database :

Here, we will discuss the queries and will understand with the help of examples.

Query-1 :

Show existing databases –

Let's consider the existing database like information\_schema, mysql, performance\_schema, sakila, student, sys, and world. And if you want to show the existing database then we will use the show database query as follows.

SHOW DATABASES;

Output :

Existing database Name

information\_schema

mysql

performance\_schema

sakila

student

sys

world

Query-2 :

Drop a database –

Suppose we want to drop the database namely student.

DROP DATABASE student;

SHOW DATABASES;

Database Name

information\_schema

mysql

performance\_schema

sakila

sys



world

Query-3 :

Create a database –

Suppose we want to create a database namely a bank.

```
CREATE DATABASE bank;
```

```
SHOW DATABASES;
```

Database Name

bank

information\_schema

mysql

performance\_schema

sakila

sys

world

Query-4 :

Using a database –

```
USE bank;
```

Query-5 :

Create a Table –

Here data type may be varchar, integer, date, etc.

```
CREATE TABLE table_name (
```

```
    column1 datatype,
```

```
    column2 datatype,
```

```
    ....
```

```
);
```

Example –

```
CREATE TABLE IF NOT EXISTS Employee (
```

```
    EmployeeID int,
```

```
    FirstName varchar(55),
```

```
    LastName varchar(55),
```

```
    Email varchar(150),
```

```
    DOB date
```

```
);
```

Query-6 :

Show tables in the same database –

```
SHOW TABLES;
```

Query-7 :

Dropping a Table –

DROP TABLE table\_name;

Query-8 :

Inserting values into an existing table –

INSERT INTO Employee

VALUES(1111,'Dipak','Bera','dipakbera@gmail.com','1994-11-22');

Query-9 :

Fetching values in a table –

SELECT \* FROM Employee;

Query-10 :

Not Null –

We can specify which column does not accept the null value when we insert a value(row) in a table. It will be done at the time of table creation.

```
CREATE TABLE table_name (  
    column1 datatype NOT NULL,  
    column2 datatype,  
    ....  
);
```

Query-11 :

Unique –

We can also specify that entries in a particular column should be unique.

```
CREATE TABLE table_name (  
    column1 datatype UNIQUE,  
    column2 datatype,  
    ....  
);
```

Example –

```
CREATE TABLE demo_table  
(  
    EmployeeID int NOT NULL UNIQUE,  
    FirstName varchar(55),  
    LastName varchar(55)  
);
```

Transactions:

A transaction in DBMS is a collection of logically related tasks. It consists of a set of operations.

A transaction is an assembly of consecutive actions or a single operation that is done by a single operator to execute tasks to access the information of a database.

For example: Consider the employee of a bank who transfers an amount Rs 1000 from the bank account of X to that of Y. This task may seem small and easy, but it involves a series of low-level operations.

Explanation of transaction

Transactions in the bank account of X

Open\_the\_account (X)

Previous\_Bal = X.bal

Current\_Bal= Previous-Bal-1000

X.bal=Current\_Bal

Close\_the\_account (X)

Transactions in the bank account of Y

Open\_the\_account (Y)

Previous\_Bal = Y.bal

Current\_Bal= Previous-Bal+1000

Y.bal=Current\_Bal

Close\_the\_account (Y)

Tasks of the transaction in DBMS

Enlisted below are the primary tasks of the transaction

**Read (X):** To read the value of X from the tables of the database and store the same in the memory buffer

**Write (X):** To write the value from the memory buffer back to the tables of the database

Now let us consider the following scenario

For example, there are two schedules in the above picture named as S1 and S2. The Serial Schedule is S2 in this case. If we talk about S1, the read operation is performed on data item R2 (A) once it is written by the T1 transaction. This way the data A works similarly as data item B during the transaction operation. The thing we want to tell the viewers is these schedules are conflict equivalent.

Also See: [Difference Between DBMS and RDBMS](#)

Why there is a need for view serializability?

Almost everyone who knows programming understands that a serial schedule causes no inconsistency due to no concurrent transactions. However, things are completely different in the case of a non-serial schedule that can cause inconsistency in the database as lots of transactions run simultaneously. It is important to ensure that the schedule is consistent when the prescribed one is a view serializable.

Many people might think it is best to go for a serial schedule instead of identifying whether a non-serial schedule is serializable. The answer to this question is a straight no as the system's resources are fully used due to concurrent applications of transactions.

Also See: [Pagination in SQL Server, MySQL and Oracle with Examples](#)

**Conflict Serializability**

This type of serializability is the same as a serial schedule where respective transactions are equipped with the same sets in chronological order without overlapping. Have a look at the below example for knowing view serializability below:

**Conflict Serializability**

Here, the Serial Schedule 2 (S2) is equipped with both the transaction that initializes the same data named R1 (A) in T1. On the other hand, W2 (B) have a write operation with the same transaction and data in T2.

It means both of these schedules are View Serializable Schedule. An amazing fact we want to tell the readers is all the conflict serializable are view serializable, but each of view serializable aren't conflict serializable.

Also See: [Explain Database Schema and its Types](#)

Why do we need conflict serializability?

Conflict serializability is greatly required in the simple algorithm for checking important things. It is essential to check the view serializability that perfectly relates to NP-full issues. It can turn to be a blessing for finding whether a non-serial schedule is conflicted or not.

R(X)

$X = X - 200$

W(X)

Let the value of X before transaction be 1000

The initial task reads the value of X from the table of the databases and stores it in the buffer space

The second task reduces X's amount by 200. The buffer now contains 800

The third task writes the value in the buffer to the table of the database. The final amount of X is 800

But owing to the failure of software, hardware, or power, the transaction might fail before completion.

For instance, If there is a failure in the debit transaction after the execution of the second task, then the value of X will remain in 1000, which will be unfavorable for the bank.

For the resolution of this issue, there are two significant transaction operations.

Rollback operation to undo the executed work

Commit for the permanent saving of the work.

### Serializability:

Serializability can be called a process used for finding the correct non-serial schedules in the database. It basically helps maintain the consistency in the database and often relates to the isolation features of a transaction. There is a solid sort of inconsistency caused by non-serial schedules in DBMS for those who didn't know.

In simple words, it helps in finding out whether the non-serial schedule is finalized or not. This post will talk about serializability in detail with a proper explanation. Go through the complete post and try to grab as much information as possible.

There are two types of serializability about which we will tell you in detail below:

#### **View-Serializability**

View Serializability can be defined as the schedule same as a serial schedule without any overlapping transactions. They are the same transactions in the two schedules that view similar data values.

#### **View Serializability**

**Database normalization** is a database schema design technique, by which an existing schema is modified to minimize redundancy and dependency of data.

Normalization split a large table into smaller tables and define relationships between them to increases the clarity in organizing data.

The words normalization and normal form refer to the structure of a database.

Normalization was developed by IBM researcher E.F. Codd In the 1970s.

Normalization increases clarity in organizing data in Databases.

Normalization of a Database is achieved by following a set of rules called 'forms' in creating the database.

The database normalization process is divided into following the normal form:

First Normal Form (1NF)

Second Normal Form (2NF)

Third Normal Form (3NF)

Boyce-Codd Normal Form (BCNF)

Fourth Normal Form (4NF)

Fifth Normal Form (5NF)

Each column is unique in 1NF.

Sample Employee table, it displays employees are working with multiple departments.

Employee	Age	Department
Melvin	32	Marketing, Sales
Edward	45	Quality Assurance
Alex	36	Human Resource

Employee table following 1NF:

Employee	Age	Department
Melvin	32	Marketing
Melvin	32	Sales
Edward	45	Quality Assurance
Alex	36	Human Resource

The entity should be considered already in 1NF, and all attributes within the entity should depend solely on the unique identifier of the entity.

Sample Products table:

productID	product	Brand
1	Monitor	Apple
2	Monitor	Samsung

All types of serializability have an importance that should be accessed correctly for finding great results. We are pretty assured you understand how important serializability is in the DBMS.

What are the major Advantages of Serializability in DBMS?

We believe it is very important that everyone should know what kind of benefits bring by serializability. That's why, our team has mentioned it in detail below:

The most significant benefit is this technique can be used quite conveniently and quickly. Additionally, there is no difficulty that one faces during the customization process in any manner.

The serialized classes have required a sort of flexibility for letting the constant evolution of object schema in an application.

You can find many technologies like RMI, Java Beans, and EJB that greatly depend on serialization.

It can also act as a technique to exchange objects between C++ and JAVA libraries with third-party vendors' help.

There is no need to be a pro for effectively understanding the prospect of serializability.

Also See: Different Sql Functions With Examples

What are the Disadvantages of serialization in DBMS?

Unfortunately, there are few disadvantages of serialization about which most people doesn't prefer to talk. We have prescribed about it in detail below:

It is strongly recommended to not using serializability with large-sized objects, especially overhead ones. The reason behind that is the garbage collection in the system can be delayed with it.

The serializable interface offers no fine-grained control over the access of objects. However, this issue can be eliminated by using the complex externalizable interface.

It isn't recommended to use for applications that required concurrent access without utilizing extra APIs.

What is the Significance of serialization in DBMS?

The main thing that proves the importance of serialization is the need to complete one transaction before beginning another one. This process ensures that the concurrent transactions are the same as one where the transactions are executed in a particular order.

The availability of serializability will make sure that proper access to the database is performed with the help of reading and writing operations. We hope you now understand how much serializability is important in DBMS.

Normalizations:

- 3 Scanner HP
- 4 Head phone JBL

Product table following 2NF:

Products Category table:

productID	product
1	Monitor
2	Scanner
3	Head phone

Brand table:

brandID	brand
1	Apple
2	Samsung
3	HP
4	JBL

Products Brand table:

pbID	productID	brandID
1	1	1
2	1	2
3	2	3
4	3	4

The entity should be considered already in 2NF, and no column entry should be dependent on any other entry (value) other than the key for the table.

If such an entity exists, move it outside into a new table.

3NF is achieved, considered as the database is normalized.

3NF and all tables in the database should be only one primary key.

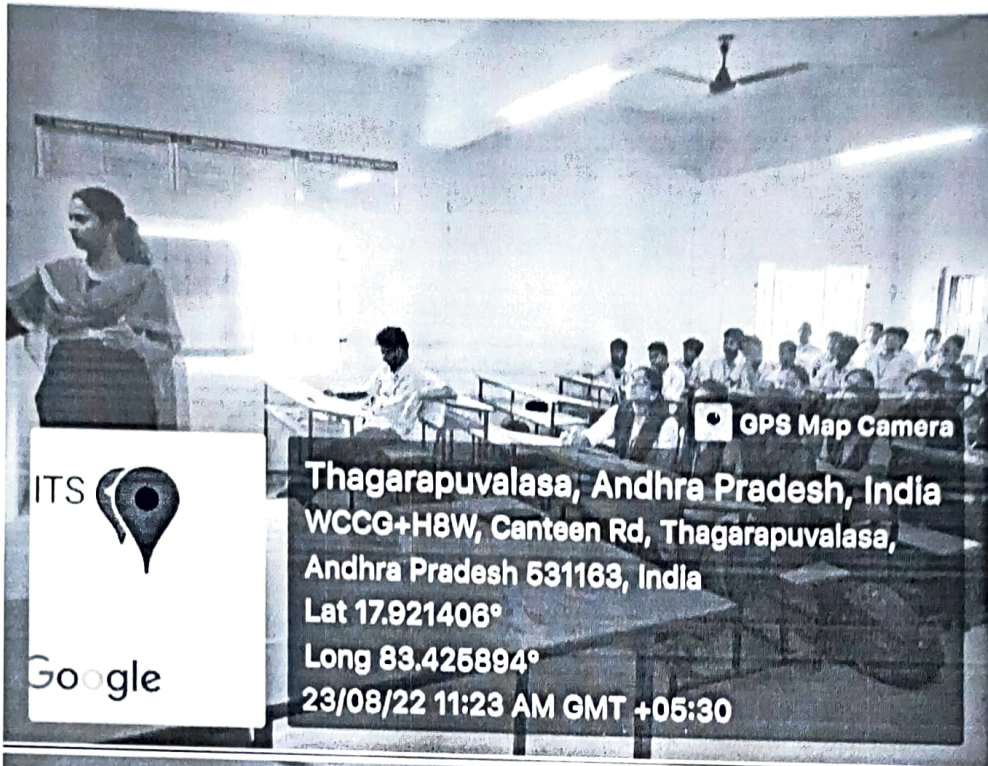
Tables cannot have multi-valued dependencies on a Primary Key.

A composite key shouldn't have any cyclic dependencies.

Well, this is a highly simplified explanation for Database Normalization. One can study this process extensively, though. After working with databases for some time, you'll automatically create Normalized databases, as it's logical and practical.

Photos:





GPS Map Camera



Google

Thagarapuvalasa, Andhra Pradesh, India  
WCCG+H8W, Canteen Rd, Thagarapuvalasa,  
Andhra Pradesh 531163, India  
Lat 17.921406°  
Long 83.425894°  
23/08/22 11:23 AM GMT +05:30



GPS Map Camera



Google

Thagarapuvalasa, Andhra Pradesh, India  
WCCG+H8W, Canteen Rd, Thagarapuvalasa,  
Andhra Pradesh 531163, India  
Lat 17.921442°  
Long 83.425867°  
23/08/22 11:23 AM GMT +05:30

Attendance:

*B. S. Jagan*  
Head of the Department of  
Computer Science & Engineering  
Andhra Pradesh State Institute of  
Technology & Sciences  
Sambalpur, Malkajgiri Dist

25/2/22

ANIL NEERUKUNDA INSTITUTE OF TECHNOLOGY AND SCIENCES (A)  
DEPARTMENT OF CSE  
3-4 B. Tech CSE - B SEM - II

GATC Class

S.No	Roll No	Name of the student	Signature
1	320126510065	ALAJANGI LIKHITHA	Alkha
2	320126510066	APPALABATHULA SATYA LOKESH	Lo
3	320126510067	ARUPAKA VINAY	A Vinay
4	320126510068	D LINA MALHURI	D Lina
5	320126510069	BALAJANGI CHAITANYA NANDA KUMAR	Chaitanya
6	320126510070	BODA VASU DEVA RAO	Vasud
7	320126510071	BONULA GOWTHAMI	Bonula
8	320126510072	BOTTA SURYA SYAM SAI SWETHAK	Surya
9	320126510073	BULUSU SRI LAKSHMI MANASA	Bulusu
10	320126510074	CHAPPA ANEESH	Chappa
11	320126510075	CHIDAM VINAY MANIKANTA KARTHIK	Chidam
12	320126510076	CHIDARLLA VIJAY KUMAR	Chidarlla
13	320126510077	CHITTALA RADHA KRISHNA NAGA SAI	Radha
14	320126510078	D VENKATA SAI HEMANTH ASWIN	D Venkata
15	320126510079	DASARI PRADEEP	Dasari
16	320126510080	DODDI LIKHITHA	Doddi
17	320126510081	DRAKSHARAPU LAUTHA VENNELA	Draksharapu
18	320126510083	GOPSETTY AKHIL VENKAT	Gopsetty
19	320126510085	GUDLA SINDHUA	Gudla
20	320126510086	GURRAM MOULI	Gurram
21	320126510087	ILLAPU ADARSH	Illapu
22	320126510088	KAKARLA AASISH PRABHU	Kakarla
23	320126510089	KANDELLI RAJ KIRAN	Kandelli
24	320126510090	KARRI VENKATA SATYA SAI RAJA REDDY	Karri
25	320126510091	KARROTHU HARI CHANDRA PRASAD	Karrothu
26	320126510092	KHANDAVALLI MANOJ	Khandavalli
27	320126510093	KODA LAKSHMI SINTRU BHARGAVI	Koda
28	320126510094	KOTANA DEEPIKA	Kotana
29	320126510095	LATCHIREDDI SAI CHARAN	Latchireddi
30	320126510096	MACHA HEMACHANDRA	Macha
31	320126510097	MARADANA TANUJA	Maradana
32	320126510098	MARAPAKULA MANASA	Marapakula
33	320126510099	MARUMUDI HADASSA	Marumudi
34	320126510100	MOHAMMED ALTAF	Mohammed
35	320126510101	MOHAN SAI NMANJUNADH	Mohan
36	320126510102	MYLIPILLI MAHESH	Mylipilli
37	320126510103	NARAYANASETTI SAILAVANYA	Narayanasetti
38	320126510104	PALADUGU RUTHVIKH	Paladugu
39	320126510105	PALAKURTHI VYSINAVI	Palakurthi
40	320126510106	PALAVALASA VINAY KUMAR	Palavalasa
41	320126510107	PASUPULETI SONI	Pasupuleti
42	320126510108	PEDDAPATI RADHA KRISHNA SWAMY	Peddapati
43	320126510109	PEDIREDLA SAI GANESH	Pediredda
44	320126510110	PENTAKOTA VIDYA MRUDULA	Pentakota
45	320126510111	PIDUGU MOUNIKA	Pidugu

S.No	Roll No	Name of the student	Signature
46	120126510112	PRODDUTURI SOURABH BHARADWAJ	
47	120126510113	PUSARLA PYDIVANKATA RAJESH	
48	120126510114	PUVVALA SIRISHA	
49	120126510115	REDDIPALLI GRUHA SATYA SAI VANAJA	
50	120126510116	ROKKAM NIKHILA	
51	120126510117	SAPPA ALEKHYA	
52	120126510118	SATTI BHENDRA VIDY NAGA MANIKANTA REDDY	
53	120126510119	SENAPATHI PRASANNA	
54	120126510120	SHAIK KHAIRUNNISA	
55	120126510121	SURISSETTY LAVANYA	
56	120126510122	THOTA CHARAN KARTHIK	
57	120126510123	VANAPARTHI JYOTHRMAI	
58	120126510124	VAVILAPALLI VINOD	
59	120126510125	VEMULAKONDA VAMSI RAMAKRISHNA	
60	120126510126	VENKATA SAI KAUSHIK ARAVELLI	
61	120126510127	VERUPANDA VAMSI	
62	120126510128	YASWANTHI BEHARA	
63	321126510L07	SHAIK MUSKAN SULTANA	
64	321126510L08	NALLI KISHORE BABU	
65	321126510L09	YALAGADA PAVAN KUMAR	
66	321126510L10	IPPILI SANJITH	
67	321126510L11	KANGANI GAYATHRI	
68	321126510L12	MULAKALAPALLI ALEKHA	

## ANIL NEERUKONDA INSTITUTE OF TECHNOLOGY AND SCIENCES (A)

DEPARTMENT OF CSE

3-4 B. Tech CSE - B SEM - II

GATE Class

23/8/22

S.No	Roll No	Name of the student	Signature
1	320126510065	ALAJANGI LIKHITHA	Likhitha
2	320126510066	APPALABATHULA SATYA LOKESH	Lokesh
3	320126510067	ARIPAKA VINAY	A. Vinay
4	320126510068	B UHA MADHURI	B. Uha Madhuri
5	320126510069	BAJJANGI CHAITANYA NANDA KUMAR	Chaitanya
6	320126510070	BODA VASU DEVA RAO	Vasudev
7	320126510071	BONULA GOWTHAMI	B. Gowthami
8	320126510072	BOTTA SURYA SYAM SAI SWETHAK	Swethak
9	320126510073	BULUSU SRI LAKSHMI MANASA	Manasa
10	320126510074	CHAPPA ANEESH	
11	320126510075	CHIDAM VINAY MANIKANTA KARTHIK	Karthik
12	320126510076	CHIDARLLA VIJAY KUMAR	Vijay
13	320126510077	CHITTALA RADHA KRISHNA NAGA SAI	Radhakrishna
14	320126510078	D VENKATA SAI HEMANTH ASWIN	
15	320126510079	DASARI PRADEEP	
16	320126510080	DODDI LIKHITHA	D. Likhitha
17	320126510081	DRAKSHARAPU LALITHA VENNELA	Vennela
18	320126510083	GOPISETTY AKHIL VENKAT	Akhil Venkat
19	320126510085	GUDLA SINDHUJA	G. Sindhuja
20	320126510086	GURRAM MOULI	Mouli
21	320126510087	ILLAPU ADARSH	Adarsh
22	320126510088	KAKARLA AASISH PRABHU	Aasish
23	320126510089	KANDELLI RAJ KIRAN	
24	320126510090	KARRI VENKATA SATYA SAI RAJA REDDY	Raja Reddy
25	320126510091	KARROTHU HARI CHANDRA PRASAD	Hari
26	320126510092	KHANDAVALLI MANOJ	Manoj
27	320126510093	KODA LAKSHMI SINDU BHARGAVI	Sindhu
28	320126510094	KOTANA DEEPIKA	
29	320126510095	LATCHIREDDI SAI CHARAN	L. Sai Charan
30	320126510096	MACHA HEMACHANDRA	Hemachandra
31	320126510097	MARADANA TANUJA	Tanuja
32	320126510098	MARAPAKULA MANASA	Manasa
33	320126510099	MARUMUDI HADASSA	Hadassa
34	320126510100	MOHAMMED ALTAF	Altaf
35	320126510101	MOHAN SAI NMANJUNADH	
36	320126510102	MYLIPILLI MAHESH	Mahesh
37	320126510103	NARAYANASETTI SAI LAVANYA	
38	320126510104	PALADUGU RUTHVIKH	Ruthvik
39	320126510105	PALAKURTHI VYSHNAVI	Vyshnavi
40	320126510106	PALAVALASA VINAY KUMAR	
41	320126510107	PASUPULETI SONI	Soni
42	320126510108	PEDDAPATI RADHA KRISHNA SWAMY	Radha
43	320126510109	PEDIREDLA SAI GANESH	Ganesh
44	320126510110	PENTAKOTA VIDYA MRUDULA	Vidya
45	320126510111	PIDUGU MOUNIKA	Mounika

S.No	Roll No	Name of the student	Signature
46	320126510112	PRODDUTURI SOURABH BHARADWAJ	<i>Sourabh</i>
47	320126510113	PUSARLA PYDI VENKATA RAJESH	<i>Rajesh</i>
48	320126510114	PUVVALA SIRISHA	<i>Sirisha</i>
49	320126510115	REDDIPALLI GRUHA SATYA SAI VANAJA	<i>Vanaja</i>
50	320126510116	ROKKAM NIKHILA	<i>Nikhila</i>
51	320126510117	SAPPA ALEKHYA	<i>Alekha</i>
52	320126510118	SATTI UHENDRA VIJAY NAGA MANIKANTA REDDY	<i>Vijay</i>
53	320126510119	SENAPATHI PRASANNA	<i>Prasanna</i>
54	320126510120	SHAIK KHAIRUNNISA	<i>Khairun</i>
55	320126510121	SURISSETTY LAVANYA	<i>Lavanya</i>
56	320126510122	THOTA CHARAN KARTHIK	<i>Charan</i>
57	320126510123	VANAPARTHI JYOTHIRMAI	<i>Jyothirmai</i>
58	320126510124	VAVILAPALLI VINOD	<i>Vinod</i>
59	320126510125	VEMULAKONDA VAMSI RAMAKRISHNA	<i>Vamsi</i>
60	320126510126	VENKATA SAI KAUSHIK ARAVELLI	<i>Kaushik</i>
61	320126510127	VERUPANDA VAMSI	<i>Vamsi</i>
62	320126510128	YASWANTH BEHARA	<i>Yaswanth</i>
63	321126510L07	SHAIK MUSKAN SULTANA	<i>Muskan</i>
64	321126510L08	NALLI KISHORE BABU	<i>Kishore</i>
65	321126510L09	YALAGADA PAVAN KUMAR	<i>Pavan</i>
66	321126510L10	IPPILI SANJITH	<i>Sanjith</i>
67	321126510L11	KANGANI GAYATHRI	<i>Gayathri</i>
68	321126510L12	MULAKALAPALLI ALEKYA	<i>Alekha</i>

*D. J. S.*  
Head of the Department of  
Computer Science & Engineering  
Anil Neerukonda Institute of  
Technology & Sciences  
Sangivalasa, Visakhapatnam Dist





**Anil Neerukonda Institute of Technology & Sciences  
(Autonomous)**

(Affiliated to AU, Approved by AICTE & Accredited by NBA (ECE, EEE, CSE, IT & Mech.) & NAAC)  
Sangmalasa- 531 162, Bheemunipatnam Mandal, Visakhapatnam District  
Phone: 08933 22 508 1/8 4/8 7 Fax: 2 263 95  
Website: www.anits.edu.in email: principal@anits.edu.in

Date: 24-08-2022

Topic: Compiler Design

Faculty: K.Amaravati, Asst Professor.

On 24-08-2022, K.Amaravati addressed the students on Data Structures related to Gate Examinations to all the students. It was an interactive session with the students where the students understood all the topics clearly.

**Topics Covered:**

- Introduction of Compiler design
- Compiler construction tools
- Phases of a Compiler
- Symbol Table in Compiler

**Abstract:**

Introduction of Compiler design

The compiler is software that converts a program written in a high-level language (Source Language) to a low-level language (Object/Target/Machine Language/0's, 1's).

Cross Compiler that runs on a machine 'A' and produces a code for another machine 'B'. It is capable of creating code for a platform other than the one on which the compiler is running. Source-to-source Compiler or transcompiler or transpiler is a compiler that translates source code written in one programming language into the source code of another programming language.

Language processing systems (using Compiler): We know a computer is a logical assembly of Software and Hardware. The hardware knows a language, that is hard for us to grasp, consequently, we tend to write programs in a high-level language, that is much less complicated for us to comprehend and maintain in thoughts. Now, these programs go through a series of transformations so that they can readily be used by machines. This is where language procedure systems come in handy.

A translator or language processor is a program that translates an input program written in a programming language into an equivalent program in another language. The compiler is a type

of translator, which takes a program written in a high-level programming language as input and translates it into an equivalent program in low-level languages such as machine language or assembly language. The program written in a high-level language is known as a source program, and the program converted into low-level language is known as an object (or target) program. Moreover, the compiler traces the errors in the source program and generates the error report. Without compilation, no program written in a high-level language can be executed. After compilation, only the program in machine language is loaded into the memory for execution. For every programming language, we have a different compiler; however, the basic tasks performed by every compiler are the same.

**High-Level Language:** If a program contains `#define` or `#include` directives such as `#include` or `#define` it is called HLL. They are closer to humans but far from machines. These (`#`) tags are called preprocessor directives. They direct the pre-processor about what to do.

**Pre-Processor:** The pre-processor removes all the `#include` directives by including the files called file inclusion and all the `#define` directives using macro expansion. It performs file inclusion, augmentation, macro-processing, etc.

**Assembly Language:** It's neither in binary form nor high level. It is an intermediate state that is a combination of machine instructions and some other useful data needed for execution.

**Assembler:** For every platform (Hardware + OS) we will have an assembler. They are not universal since for each platform we have one. The output of the assembler is called an object file. Its translates assembly language to machine code.

**Interpreter:** An interpreter converts high-level language into low-level machine language, just like a compiler. But they are different in the way they read the input. The Compiler in one go reads the inputs, does the processing, and executes the source code whereas the interpreter does the same line by line. A compiler scans the entire program and translates it as a whole into machine code whereas an interpreter translates the program one statement at a time.

Interpreted programs are usually slower with respect to compiled ones.

**Relocatable Machine Code:** It can be loaded at any point and can be run. The address within the program will be in such a way that it will cooperate with the program movement.

**Loader/Linker:** It converts the relocatable code into absolute code and tries to run the program resulting in a running program or an error message (or sometimes both can happen). Linker loads a variety of object files into a single file to make it executable. Then loader loads it in memory and executes it.

**Phases of a Compiler:**

There are two major phases of compilation, which in turn have many parts. Each of them takes input from the output of the previous level and works in a coordinated way.

**Analysis Phase:** An intermediate representation is created from the given source code :

Lexical Analyzer

Syntax Analyzer

Semantic Analyzer

Intermediate Code Generator

Lexical analyzer divides the program into "tokens", the Syntax analyzer recognizes "sentences" in the program using the syntax of the language and the Semantic analyzer checks the static semantics of each construct. Intermediate Code Generator generates "abstract" code.

**Synthesis Phase:** Equivalent target program is created from the intermediate representation. It has two parts :

**Code Optimizer**

**Code Generator**

Code Optimizer optimizes the abstract code, and the final Code Generator translates abstract intermediate code into specific machine instructions.

#### Compiler construction tools:

The compiler writer can use some specialized tools that help in implementing various phases of a compiler. These tools assist in the creation of an entire compiler or its parts. Some commonly used compiler construction tools include:

**Parser Generator –**

It produces syntax analyzers (parsers) from the input that is based on a grammatical description of programming language or on a context-free grammar. It is useful as the syntax analysis phase is highly complex and consumes more manual and compilation time.

Example: PIC, EQM

**Scanner Generator –**

It generates lexical analyzers from the input that consists of regular expression description based on tokens of a language. It generates a finite automaton to recognize the regular expression.

Example: Lex

**Syntax directed translation engines –**

It generates intermediate code with three address format from the input that consists of a parse tree. These engines have routines to traverse the parse tree and then produces the intermediate code. In this, each node of the parse tree is associated with one or more translations.

**Automatic code generators –**

It generates the machine language for a target machine. Each operation of the intermediate language is translated using a collection of rules and then is taken as an input by the code generator. A template matching process is used. An intermediate language statement is replaced by its equivalent machine language statement using templates.

**Data-flow analysis engines –**



It is used in code optimization. Data flow analysis is a key part of the code optimization that gathers the information, that is the values that flow from one part of a program to another. Refer – data flow analysis in Compiler

Compiler construction toolkits –

It provides an integrated set of routines that aids in building compiler components or in the construction of various phases of compiler.

### Phases of a Compiler:

We basically have two phases of compilers, namely the Analysis phase and Synthesis phase. The analysis phase creates an intermediate representation from the given source code. The synthesis phase creates an equivalent target program from the intermediate representation.

Symbol Table – It is a data structure being used and maintained by the compiler, consisting of all the identifier's names along with their types. It helps the compiler to function smoothly by finding the identifiers quickly.

The analysis of a source program is divided into mainly three phases. They are:

Linear Analysis-

This involves a scanning phase where the stream of characters is read from left to right. It is then grouped into various tokens having a collective meaning.

Hierarchical Analysis-

In this analysis phase, based on a collective meaning, the tokens are categorized hierarchically into nested groups.

Semantic Analysis-

This phase is used to check whether the components of the source program are meaningful or not.

The compiler has two modules namely the front end and the back end. Front-end constitutes the Lexical analyzer, semantic analyzer, syntax analyzer, and intermediate code generator. And the rest are assembled to form the back end.

Lexical Analyzer –

It is also called a scanner. It takes the output of the preprocessor (which performs file inclusion and macro expansion) as the input which is in a pure high-level language. It reads the characters from the source program and groups them into lexemes (sequence of characters that "go together"). Each lexeme corresponds to a token. Tokens are defined by regular expressions which are understood by the lexical analyzer. It also removes lexical errors (e.g., erroneous characters), comments, and white space.

Syntax Analyzer – It is sometimes called a parser. It constructs the parse tree. It takes all the tokens one by one and uses Context-Free Grammar to construct the parse tree.

Why Grammar?

The rules of programming can be entirely represented in a few productions. Using these productions we can represent what the program actually is. The input has to be checked whether it is in the desired format or not.

**Syntax Analysis:** Adds information regarding attribute type, scope, dimension, line of reference, use, etc in the table.

**Semantic Analysis:** Uses available information in the table to check for semantics i.e. to verify that expressions and assignments are semantically correct (type checking) and update it accordingly.

**Intermediate Code generation:** Refers symbol table for knowing how much and what type of run-time is allocated and table helps in adding temporary variable information.

**Code Optimization:** Uses information present in the symbol table for machine-dependent optimization.

**Target Code generation:** Generates code by using address information of identifier present in the table.

**Symbol Table entries** – Each entry in the symbol table is associated with attributes that support the compiler in different phases.

Items stored in Symbol table:

- Variable names and constants
- Procedure and function names
- Literal constants and strings
- Compiler generated temporaries
- Labels in source languages
- Information used by the compiler from Symbol table:

Data type and name

Declaring procedures

Offset in storage

If structure or record then, a pointer to structure table.

For parameters, whether parameter passing by value or by reference

Number and type of arguments passed to function

Base Address

**Operations of Symbol table** – The basic operations defined on a symbol table include:

**Implementation of Symbol table** –

Following are commonly used data structures for implementing symbol table:-

**List** –

In this method, an array is used to store names and associated information.

A pointer "available" is maintained at end of all stored records and new names are added in the order as they arrive

To search for a name we start from the beginning of the list till available pointer and if not found we get an error "use of the undeclared name"

While inserting a new name we must ensure that it is not already present otherwise an error occurs i.e. "Multiple defined names"

Insertion is fast  $O(1)$ , but lookup is slow for large tables –  $O(n)$  on average

The parse tree is also called the derivation tree. Parse trees are generally constructed to check for ambiguity in the given grammar. There are certain rules associated with the derivation tree.

Any identifier is an expression

Any number can be called an expression

Performing any operations in the given expression will always result in an expression. For example, the sum of two expressions is also an expression.

The parse tree can be compressed to form a syntax tree

Syntax error can be detected at this level if the input is not in accordance with the grammar.

**Semantic Analyzer** – It verifies the parse tree, whether it's meaningful or not. It furthermore produces a verified parse tree. It also does type checking, Label checking, and Flow control checking.

**Intermediate Code Generator** – It generates intermediate code, which is a form that can be readily executed by a machine. We have many popular intermediate codes. Example – Three address codes etc. Intermediate code is converted to machine language using the last two phases which are platform-dependent.

Till intermediate code, it is the same for every compiler out there, but after that, it depends on the platform. To build a new compiler we don't need to build it from scratch. We can take the intermediate code from the already existing compiler and build the last two parts.

**Code Optimizer** – It transforms the code so that it consumes fewer resources and produces more speed. The meaning of the code being transformed is not altered. Optimization can be categorized into two types: machine-dependent and machine-independent.

**Target Code Generator** – The main purpose of the Target Code generator is to write a code that the machine can understand and also register allocation, instruction selection, etc. The output is dependent on the type of assembler. This is the final stage of the compilation. The optimized code is converted into relocatable machine code which then forms the input to the linker and loader.

### Symbol Table:

Symbol Table is an important data structure created and maintained by the compiler in order to keep track of semantics of variables i.e. it stores information about the scope and binding information about names, information about instances of various entities such as variable and function names, classes, objects, etc.

It is built-in lexical and syntax analysis phases.

The information is collected by the analysis phases of the compiler and is used by the synthesis phases of the compiler to generate code.

It is used by the compiler to achieve compile-time efficiency.

It is used by various phases of the compiler as follows:-

**Lexical Analysis:** Creates new table entries in the table, for example like entries about tokens.

The advantage is that it takes a minimum amount of space.

#### Linked List –

This implementation is using a linked list. A link field is added to each record.

Searching of names is done in order pointed by the link of the link field.

A pointer "First" is maintained to point to the first record of the symbol table.

Insertion is fast  $O(1)$ , but lookup is slow for large tables –  $O(n)$  on average

#### Hash Table –

In hashing scheme, two tables are maintained – a hash table and symbol table and are the most commonly used method to implement symbol tables.

A hash table is an array with an index range: 0 to table size – 1. These entries are pointers pointing to the names of the symbol table.

To search for a name we use a hash function that will result in an integer between 0 to table size – 1.

Insertion and lookup can be made very fast –  $O(1)$ .

The advantage is quick to search is possible and the disadvantage is that hashing is complicated to implement.

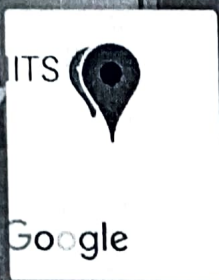
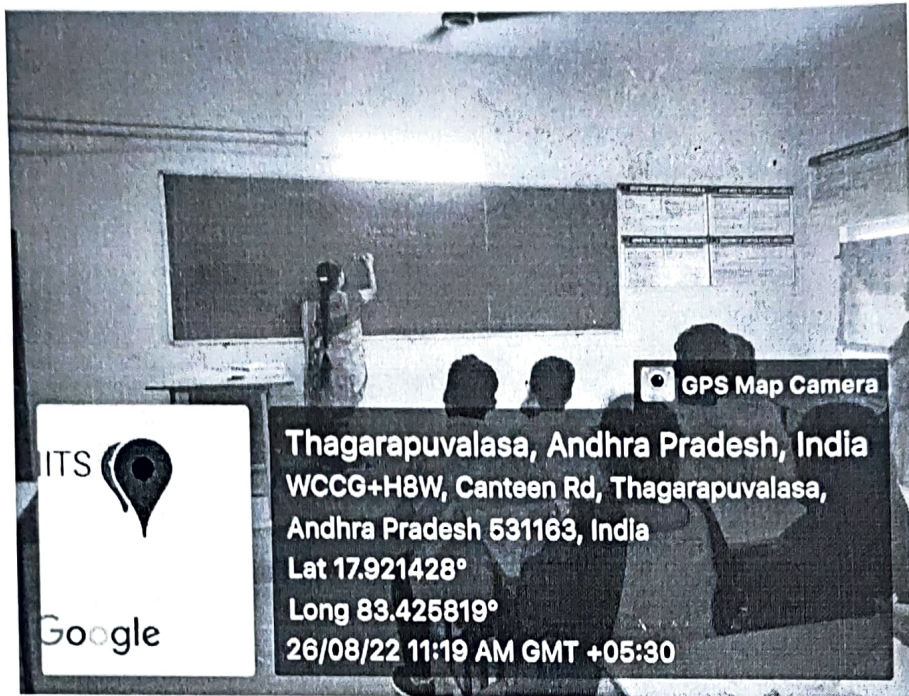
#### Binary Search Tree –

Another approach to implementing a symbol table is to use a binary search tree i.e. we add two link fields i.e. left and right child.

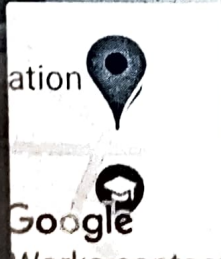
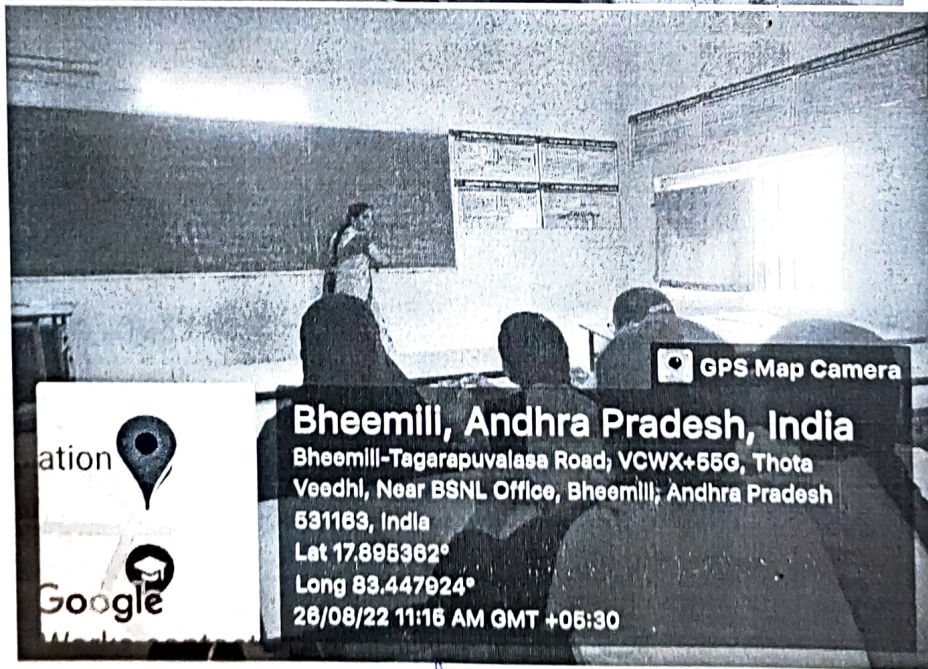
All names are created as child of the root node that always follows the property of the binary search tree.

Insertion and lookup are  $O(\log_2 n)$  on average.

Photos:



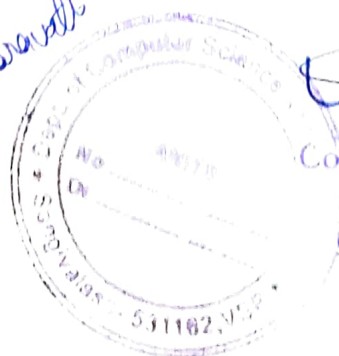
**Thagarapuvalasa, Andhra Pradesh, India**  
 WCCG+H8W, Canteen Rd, Thagarapuvalasa,  
 Andhra Pradesh 531163, India  
 Lat 17.921428°  
 Long 83.425819°  
 26/08/22 11:19 AM GMT +05:30



**Bheemili, Andhra Pradesh, India**  
 Bheemili-Tagarapuvalasa Road, VCWX+55G, Thota  
 Veedhi, Near BSNL Office, Bheemili, Andhra Pradesh  
 531163, India  
 Lat 17.895362°  
 Long 83.447924°  
 26/08/22 11:16 AM GMT +05:30

*Attendance*  
**Attendance**

*Attenuation*



*Pradeep Kumar*  
 Head of the Department of  
 Computer Science & Engineering  
 Anil Neerunde Institute of  
 Technology & Sciences  
 Sangivalasa, Visakhapatnam Dist

## ANIL NEERUKONDA INSTITUTE OF TECHNOLOGY AND SCIENCES (A)


## DEPARTMENT OF CSE

## 2-4 B. Tech CSE - B SEM - II

Gate Passes

S.No	Roll No	Name of the student	Signature
1	320126510065	ALAJANGI LIKHITHA	<i>Alajangi Likhitha</i>
2	320126510066	APPALABATHULA SATYA LOKESH	<i>Appalabathula Satya Lokesh</i>
3	320126510067	ARIPAKA VINAY	<i>Aripaka Vinay</i>
4	320126510068	B UHA MADHURI	<i>B Uha Madhuri</i>
5	320126510069	BAJJANGI CHAITANYA NANDA KUMAR	<i>Bajjangi Chaitanya Nanda Kumar</i>
6	320126510070	BODA VASU DEVA RAO	<i>Boda Vasu Deva Rao</i>
7	320126510071	BONULA GOWTHAMI	<i>Bonula Gowthami</i>
8	320126510072	BOTTA SURYA SYAM SAI SWETHAK	<i>Botta Surya Syam Sai Swethak</i>
9	320126510073	BULUSU SRI LAKSHMI MANASA	<i>Bulusu Sri Lakshmi Manasa</i>
10	320126510074	CHAPPA ANEESH	
11	320126510075	CHIDAM VINAY MANIKANTA KARTHIK	<i>Chidam Vinay Manikanta Karthik</i>
12	320126510076	CHIDARLLA VIJAY KUMAR	<i>Chidarlla Vijay Kumar</i>
13	320126510077	CHITTALA RADHA KRISHNA NAGA SAI	<i>Chittala Radha Krishna Naga Sai</i>
14	320126510078	D VENKATA SAI HEMANTH ASWIN	
15	320126510079	DASARI PRADEEP	
16	320126510080	DODDI LIKHITHA	<i>Doddi Likhitha</i>
17	320126510081	DRAKSHARAPU LALITHA VENNELA	<i>Draksharapu Lalitha Vennela</i>
18	320126510083	GOPISETTY AKHIL VENKAT	<i>Gopisetty Akhil Venkat</i>
19	320126510085	GUDLA SINDHUJA	<i>Gudla Sindhuja</i>
20	320126510086	GURRAM MOULI	<i>Gurram Mouli</i>
21	320126510087	ILLAPU ADARSH	<i>Illapu Adarsh</i>
22	320126510088	KAKARLA AASISH PRABHU	<i>Kakarla Aasish Prabhu</i>
23	320126510089	KANDELLI RAJ KIRAN	
24	320126510090	KARRI VENKATA SATYA SAI RAJA REDDY	<i>Karri Venkata Satya Sai Raja Reddy</i>
25	320126510091	KARROTHU HARI CHANDRA PRASAD	<i>Karrothu Hari Chandra Prasad</i>
26	320126510092	KHANDAVALLI MANOJ	<i>Khandavalli Manoj</i>
27	320126510093	KODA LAKSHMI SINDU BHARGAVI	<i>Koda Lakshmi Sindu Bhargavi</i>
28	320126510094	KOTANA DEEPIKA	
29	320126510095	LATCHIREDDI SAI CHARAN	<i>Latchireddi Sai Charan</i>
30	320126510096	MACHA HEMACHANDRA	<i>Macha Hemachandra</i>
31	320126510097	MARADANA TANUJA	<i>Maradana Tanuja</i>
32	320126510098	MARAPAKULA MANASA	<i>Marapakula Manasa</i>
33	320126510099	MARUMUDI HADASSA	<i>Marumudi Hadassa</i>
34	320126510100	MOHAMMED ALTAF	<i>Mohammed Altaf</i>
35	320126510101	MOHAN SAI NMANJUNADH	
36	320126510102	MYLIPILLI MAHESH	<i>Mylipilli Mahesh</i>
37	320126510103	NARAYANASETTI SAI LAVANYA	
38	320126510104	PALADUGU RUTHVIKH	<i>Paladugu Ruthvikh</i>
39	320126510105	PALAKURTHI VYSHNAVI	<i>Palakurthi Vyshnavi</i>
40	320126510106	PALAVALASA VINAY KUMAR	
41	320126510107	PASUPULETI SONI	<i>Pasupuleti Soni</i>
42	320126510108	PEDDAPATI RADHA KRISHNA SWAMY	<i>Peddapati Radha Krishna Swamy</i>
43	320126510109	PEDIREDLA SAI GANESH	<i>Pediredla Sai Ganesh</i>
44	320126510110	PENTAKOTA VIDYA MRUDULA	<i>Pentakota Vidya Mrudula</i>
45	320126510111	PIDUGU MOUNIKA	<i>Pidugu Mounika</i>

S.No	Roll No	Name of the student	Signature
46	120126510112	PRODDUTURI SOURABH BIHARADWAJ	Sourabh
47	120126510113	PUNARI A PYDI VENKATA RAJESH	Venky
48	120126510114	PUVVALA SIRISHA	Sirisha
49	120126510115	REDDIPALLI GRUHA SATYA SAI VANAJA	Venky
50	120126510116	ROKKAM NIKHILA	Nikhila
51	120126510117	SAPPA ALEKHYA	Alekya
52	120126510118	NAITHUENDRA VIJAY NAGA MANIKANTA REDDY	Vijay
53	120126510119	SENAPATHI PRASANNA	Prasanna
54	120126510120	SHAIK KHAIRUNNISA	Khairunisa
55	120126510121	SURISSETTY LAVANYA	Lavanya
56	120126510122	THOTA CHARAN KARTHIK	Charan
57	120126510123	VANAPARTHI JYOTHIRMAI	Jyothirmai
58	120126510124	VAVILAPALLI VINOD	Vinod
59	120126510125	VEMULAKONDA VAMSI RAMAKRISHNA	Vamsi
60	120126510126	VENKATA SAI KAUSHIK ARAVELLI	Kaushik
61	120126510127	VERUPANDA VAMSI	Vamsi
62	120126510128	YASWANTHI BEHARA	Yaswanthi
63	321126510L07	SHAIK MUSKAN SULTANA	Muskan
64	321126510L08	NALLI KISHORE BABU	Kishore
65	321126510L09	YALAGADA PAVAN KUMAR	Pavan
66	321126510L10	IPPILI SANJITH	Sanjith
67	321126510L11	KANGANI GAYATHRI	Gayathri
68	321126510L12	MULAKALAPALLI ALEKYA	Alekya

  
 Head of the Department of  
 Computer Science & Engineering  
 Anil Neerukonda Institute of  
 Technology & Sciences  
 Sanjivalasa, Visakhapatnam Dist



## ANIL NEERUKONDA INSTITUTE OF TECHNOLOGY AND SCIENCES (A)

## DEPARTMENT OF CSE

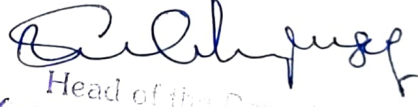
## 24 B. Tech CSE - B SEM - II

24/8/22  
Gate Classes

S.No	Roll No	Name of the student	Signature
1	320126510065	ALAJANGI LIKHITHA	Alakhitha
2	320126510066	APPALABATHULA SATYA LOKESH	Satya Lokesh
3	320126510067	ARIPAKA VINAY	A Vinay
4	320126510068	B UHA MADHURI	B. U. Madhuri
5	320126510069	BAJJANGI CHAITANYA NANDA KUMAR	Chaitanya
6	320126510070	BODA VASU DEVA RAO	Vasudeva
7	320126510071	BONULA GOWTHAMI	B. Gowthami
8	320126510072	BOTTA SURYA SYAM SAI SWETHAK	Swethak
9	320126510073	BULUSU SRI LAKSHMI MANASA	Manasa
10	320126510074	CHAPPA ANEESH	
11	320126510075	CHIDAM VINAY MANIKANTA KARTHIK	Karthik
12	320126510076	CHIDARLLA VIJAY KUMAR	Vijay
13	320126510077	CHITTALA RADHA KRISHNA NAGA SAI	Radha Krishna
14	320126510078	D VENKATA SAI HEMANTH ASWIN	
15	320126510079	DASARI PRADEEP	
16	320126510080	DODDI LIKHITHA	D. Likhitha
17	320126510081	DRAKSHARAPU LALITHA VENNELA	Vennela
18	320126510083	GOPSETTY AKHIL VENKAT	Akhil Venkat
19	320126510085	GUDLA SINDHUJA	G. Sindhuja
20	320126510086	GURRAM MOULI	Mouli
21	320126510087	ILLAPU ADARSH	Adarsh
22	320126510088	KAKARLA AASISH PRABHU	Aashish
23	320126510089	KANDELLI RAJ KIRAN	
24	320126510090	KARRI VENKATA SATYA SAI RAJA REDDY	Raja Reddy
25	320126510091	KARROTHU HARI CHANDRA PRASAD	Hari
26	320126510092	KHANDAVALLI MANOJ	Manoj
27	320126510093	KODA LAKSHMI SINDU BHARGAVI	Sindhu
28	320126510094	KOTANA DEEPIKA	
29	320126510095	LATCHIREDDI SAI CHARAN	L. Sai Charan
30	320126510096	MACHA HEMACHANDRA	Hemachandra
31	320126510097	MARADANA TANUJA	Tanuja
32	320126510098	MARAPAKULA MANASA	Manasa
33	320126510099	MARUMUDI HADASSA	Hadassa
34	320126510100	MOHAMMED ALTAF	Altat
35	320126510101	MOHAN SAI NMANJUNADH	
36	320126510102	MYLIPILLI MAHESH	Mahesh
37	320126510103	NARAYANASETTI SAI LAVANYA	
38	320126510104	PALADUGU RUTHVIKH	R. Ruthvik
39	320126510105	PALAKURTHI VYSHNAVI	Vyshu
40	320126510106	PALAVALASA VINAY KUMAR	
41	320126510107	PASUPULETI SONI	Soni
42	320126510108	PEDDAPATI RADHA KRISHNA SWAMY	Radha
43	320126510109	PEDIREDLA SAI GANESH	Ganesh
44	320126510110	PENTAKOTA VIDYA MRUDULA	Mrudula
45	320126510111	PIDUGU MOUNIKA	Mounika



S.No	Roll No	Name of the student	Signature
46	320126510112	PRODDUTURI SOURABH BHARADWAJ	Sourabh
47	320126510113	PUSARLA PYDI VENKATA RAJESH	Rajesh
48	320126510114	PUVVALA SIRISHA	Sirisha
49	320126510115	REDDIPALLI GRUHA SATYA SAI VANAJA	Vanaja
50	320126510116	ROKKAM NIKHILA	Nikhila
51	320126510117	SAPPA ALEKHYA	Alekhy
52	320126510118	SATTI UHENDRA VIJAY NAGA MANIKANTA REDDY	Manikanta
53	320126510119	SENAPATHI PRASANNA	Prasanna
54	320126510120	SHAIK KHAIRUNNISA	Khairun
55	320126510121	SURISSETTY LAVANYA	Lavanya
56	320126510122	THOTA CHARAN KARTHIK	Charan
57	320126510123	VANAPARTHI JYOTHIRMAI	Jyothirmai
58	320126510124	VAVILAPALLI VINOD	Vinod
59	320126510125	VEMULAKONDA VAMSI RAMAKRISHNA	Vamsi
60	320126510126	VENKATA SAI KAUSHIK ARAVELLI	Kaushik
61	320126510127	VERUPANDA VAMSI	Vamsi
62	320126510128	YASWANTH BEHARA	Yaswanth
63	321126510L07	SHAIK MUSKAN SULTANA	Muskan
64	321126510L08	NALLI KISHORE BABU	Kishore
65	321126510L09	YALAGADA PAVAN KUMAR	Pavan
66	321126510L10	IPPILI SANJITH	Sanjith
67	321126510L11	KANGANI GAYATHRI	Gayathri
68	321126510L12	MULAKALAPALLI ALEKYA	Alekya

  
 Head of the Department of  
 Computer Science & Engineering  
 Anil Neerukonda Institute of  
 Technology & Sciences  
 Sangivalasa, Visakhapatnam Dist







**Anil Neerukonda Institute of Technology & Sciences  
(Autonomous)**

(Affiliated to AU, Approved by AICTE & Accredited by NBA (ECE, EEE, CSE, IT & Mech) & NAAC)  
Sangivalasa- 531 162, Bheemurupatnam Mandal, Visakhapatnam District  
Phone: 08 933 22 508 3/8 4/8 7 Fax: 2 263 95  
Website: www.anits.edu.in email: principal@anits.edu.in

Date: 25-08-2022

Topic: Data Structures  
Faculty: D. Ashwini, Asst Professor

On 25-08-2022, D. Ashwini addressed the students on Data Structures related to Gate Examinations to all the students. It was an interactive session with the students where the students understood all the topics clearly.

Abstract:

- Linked Lists Notes
- Graphs and their Applications
- Trees
- Stacks and their Applications
- Queues Notes

Linked List:

A linked list is a sequence of data structures. It is also known as a linear data structure that comprises a set of connected nodes. Each node is used to store the data and also the address of the next node.

Explanation of Picture

The starting point of the linked list is known as the head of the list. It is not a different node, but refers to the first node.

The node present in the end is called NULL.

Types of Linked List

There are 3 different types of Linked Lists:

Singly Linked List

Doubly Linked List

Circular Linked List

1. Single Linked List

It is the most manageable type of linked list in which every node includes some data and the address part, which means a pointer to the next node in the series. In a singly linked list, we can perform operations like insertion, deletion, and traversal.

## 2. Doubly Linked List

When a node holds a data part and two addresses, it is known as a doubly-linked list. Two addresses means a pointer to the previous node and the next node.

## 3. Circular Linked List

In a circular linked list, the last node of the series contains the address of the first node to make a circular chain.

### Basic Operations In Linked List

When it comes to linked lists, there are five operations supported by the series or list.

**Traversal** – Through this operation, we can access elements.

**Insertion** – In this operation, elements can be added at the starting of the list.

**Deletion** – In this operation, elements can be deleted at the starting of the list.

**Search** – Through this operation, we can easily search for an element utilising the provided key.

**Sort** – Through this operation, we can sort the nodes of the linked list.

### Advantages of Linked Lists

A linked list is dynamic, which means it will provide memory whenever needed.

In a linked list, we can swiftly execute the operations like insertion and deletion.

We can easily implement stacks and queues.

It helps in reducing the access time.

### Disadvantages of Linked Lists

Sometimes the memory gets wasted because pointers need extra memory for storage.

We can access elements in sequence. You cannot do this process in a random manner.

In a linked list, reverse traversing is challenging.

### Trees:

A tree is a non-linear and hierarchical data structure that has a group of nodes. When it comes to the tree, each node stores a value.

### Important Terminologies in Tree

**Node:** A node is an entity that contains a key or value and pointers to its child nodes.

**Tree Node**

**Edge:** The connection between any two nodes is known as the edge.

**Tree Edge**

**Root:** The topmost node of a tree is known as the root.

**Tree Root**

**Complete Figure That Explains Tree Terminologies**

**Complete Tree**

### Types of Trees

Binary Tree

Binary Search Tree

AVL Tree

B-Tree

### Properties of Trees

A tree is a hierarchical structure as it contains multiple levels.

In a tree, the topmost node is known as the root node.

A node that doesn't have a child node is known as a leaf node or terminal node.

The highest number of nodes at every level of  $i$  is  $2^i$ .

Height of the tree = the longest path from the root node to the leaf node.

Depth of a node = the length of the path to its root.

### Tree Applications

Trees store data in the form of a hierarchical structure.

The tree is used to manage the data efficiently. It improves the following process like insertion, deletion and searching.

Tree is also utilised to keep the data in routing tables in the routers.

### Stacks:

A Stack is a linear data structure that holds a linear, ordered sequence of elements. It is an abstract data type. A Stack works on the LIFO process (Last In First Out), i.e., the element that was inserted last will be removed first. To implement the Stack, it is required to maintain a pointer to the top of the Stack, which is the last element to be inserted because we can access the elements only on the top of the Stack.

**Stack**

## Last in first out in Stack

### Operation on Stack

#### Push and Pop Operation in Stack

1. **PUSH:** PUSH operation implies the insertion of a new element into a Stack. A new element is always inserted from the topmost position of the Stack; thus, we always need to check if the top is empty or not, i.e.,  $TOP = \text{Max} - 1$  if this condition goes false, it means the Stack is full, and no more elements can be inserted, and even if we try to insert the element, a Stack overflow message will be displayed.

Algorithm:

Step-1: If  $TOP = \text{Max} - 1$

Print "Overflow"

Goto Step 4

Step-2: Set  $TOP = TOP + 1$

Step-3: Set  $\text{Stack}[TOP] = \text{ELEMENT}$

Step-4: END

2. **POP:** POP means to delete an element from the Stack. Before deleting an element, make sure to check if the Stack Top is NULL, i.e.,  $TOP = \text{NULL}$ . If this condition goes true, it means the Stack is empty, and no deletion operation can be performed, and even if we try to delete, then the Stack underflow message will be generated.

Algorithm:

Step-1: If  $TOP = \text{NULL}$

Print "Underflow"

Goto Step 4

Step-2: Set  $VAL = \text{Stack}[TOP]$

Step-3: Set  $TOP = TOP - 1$

Step-4: END

3. PEEK: When we need to return the value of the topmost element of the Stack without deleting it from the Stack, the Peek operation is used. This operation first checks if the Stack is empty, i.e., TOP = NULL; if it is so, then an appropriate message will display, else the value will return.

Algorithm:

Step-1: If TOP = NULL

PRINT "Stack is Empty"

Goto Step 3

Step-2: Return Stack[TOP]

Step-3: END

Representation of the Stack

A Stack can be a fixed specific size, or it can be dynamic, i.e., the Stack size can be changed dynamically. It can be represented by means of Pointer, Array, Structure, and Linked List.

Representation of the Stack

Application of the Stack

A Stack can be used for evaluating expressions consisting of operands and operators. Stacks can be used for Backtracking, i.e., to check parenthesis matching in an expression. It can also be used to convert one form of expression to another form. It can be used for systematic Memory Management.

Advantages of Stack

A Stack helps to manage the data in the 'Last in First out' method. When the variable is not used outside the function in any program, the Stack can be used. It allows you to control and handle memory allocation and deallocation. It helps to automatically clean up the objects.

Disadvantages of Stack

It is difficult in Stack to create many objects as it increases the risk of the Stack overflow. It has very limited memory.

## Graphs.

**Graph:** A graph is defined as a collection of nodes (known as vertices (in a graph)) and connections between them (known as edges).

**Directed Graph:** In graph theory, a directed graph is defined as a graph whose edges have direction, then this kind of graph is known as a directed graph or digraph and the edges with direction are known as directed edges or arcs.

**Adjacency:** If two nodes  $a, b$  are connected by an edge in the graph then we say  $a$  is adjacent to  $b$ .

**Path:** A path in a graph is defined as a finite or infinite sequence of edges that join a sequence of nodes (vertices).

**Loop:** A loop may be defined as a path that has same start and end vertex. Or we can say that a path that starts and ends on the same vertex

**Connected Graph:** In a connected graph there exists a path between every pair of vertices, no node is disconnected.

**Acyclic Graph:** Acyclic graph is a graph with no cycles.

**Weighted Graphs:** A weighted graph is defined as a graph whose each edge have some weight (value).

**Weight of a Graph:** Sum of weights(values) of all edges in a graph is known as weight of the graph.

**Connected Components:** In an undirected graph, a connected component is defined as a subset of vertices which are mostly reachable from one another. A graph having one connected component known as connected graph, i.e. each vertex is reachable from another.

**Subgraph:** Subgraph is defined as the subset of vertices and edges that forms a graph.

**Tree:** Tree is a connected graph without any cycle and loop.

**Forest:** A forest may be defined as an undirected and disconnected, acyclic graph. A disjoint collection of trees is also known as forest. Each component of a forest is tree.

**Weakly Connected component:** Connected graph which is not strongly connected, i.e some vertex are not connected to every vertex known as weakly connected graph.

**Graph Representations:** There are numerous ways of representing a graph:

Incidence list  
Incidence matrix  
Adjacency List  
Adjacency Matrix

**Graph Traversals.** Graph traversals defined as a process of traversing (or searching, reaching) all vertices in graph. If the graph is connected then only we can reach to all vertices of graph. While traversing we will never visit a vertex more than once.

To search a node (or traverse a node) we use **bread first search (bfs)** and **depth first search (dfs)** algorithm.

**Depth first search (DFS) Algorithm**

**Step 1:** Visit one vertex, you can choose any vertex as the starting node (if start vertex not mentioned in question). And push that vertex to the Stack.

**Step 2:** find another undiscovered adjacent nodes of the top element of the stack and visit any one of them (in any particular order).

**Step 3:** Repeat Step 2 until there is no undiscovered vertex(nodes) left.

**Step 4:** Pop out the element from the top of the Stack and Repeat Step 2, 3 and 4 till the stack becomes empty.

By using DFS traversal, only some edges will be traversed and these edges will form a tree, which will known as **depth-first-search tree** of graph  $G$ , and the edges of this tree are known as **tree edges**. The edges of  $G$  can be described into three parts:

**Back edges** points a child node to one of its ancestors in the DFS tree.

**Forward edges** point from a node to one of its child or descendants.

**Cross edges** point from a node to a previously visited node that is neither its ancestor nor descendant.

**Applications of DFS**

To find strongly connected components of graph

To find bridges in graph

Minimum spanning tree

To check if graph has a cycle

Topological sorting

**Analysis of the DFS:** The running time complexity of the DFS algorithm is  $O(|V|+|E|)$

**Breadth First Search Algorithm**

**Step 1:** Visit the start vertex (node), you can choose any node as the start node(if it is not given in the question). And add that node into a queue.



Step 2: Repeat the below steps until the queue becomes empty.

Step 3: Remove the head element of the queue and while staying at the vertex, visit all connected vertices and then add them into queue one by one (choose any order to visit all the connected vertices).

Step 4: When you visit all the connected vertices. Repeat Step 3.

#### Applications of BFS

To find all nodes within one connected component

To check if graph has a cycle

Diameter of Tree

To find shortest path between two nodes  $u$  and  $v$

To test bipartite-ness of a graph

Analysis of BFS: The running time complexity of the BFS algorithm is  $O(|V|+|E|)$ .

#### Graph Applications:

Electronic circuits

Task scheduling

Route mapping

Packet routing in Networks

#### Queues:

A queue is defined as a linear data structure that is open at both ends and the operations are performed in First In First Out (FIFO) order.

We define a queue to be a list in which all additions to the list are made at one end, and all deletions from the list are made at the other end. The element which is first pushed into the order, the operation is first performed on that.

#### What is Queue in Data Structure and Algorithms

##### FIFO Principle of Queue:

A Queue is like a line waiting to purchase tickets, where the first person in line is the first person served. (i.e. First come first serve).

Position of the entry in a queue ready to be served, that is, the first entry that will be removed from the queue, is called the front of the queue (sometimes, head of the queue), similarly, the position of the last entry in the queue, that is, the one most recently added, is called the rear (or the tail) of the queue. See the below figure.

FIFO property of queue

FIFO property of queue

##### Characteristics of Queue:

Queue can handle multiple data.

We can access both ends.

They are fast and flexible.

Queue Representation:

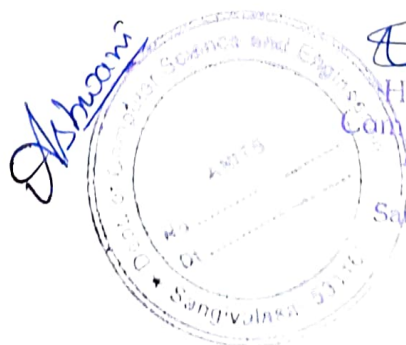
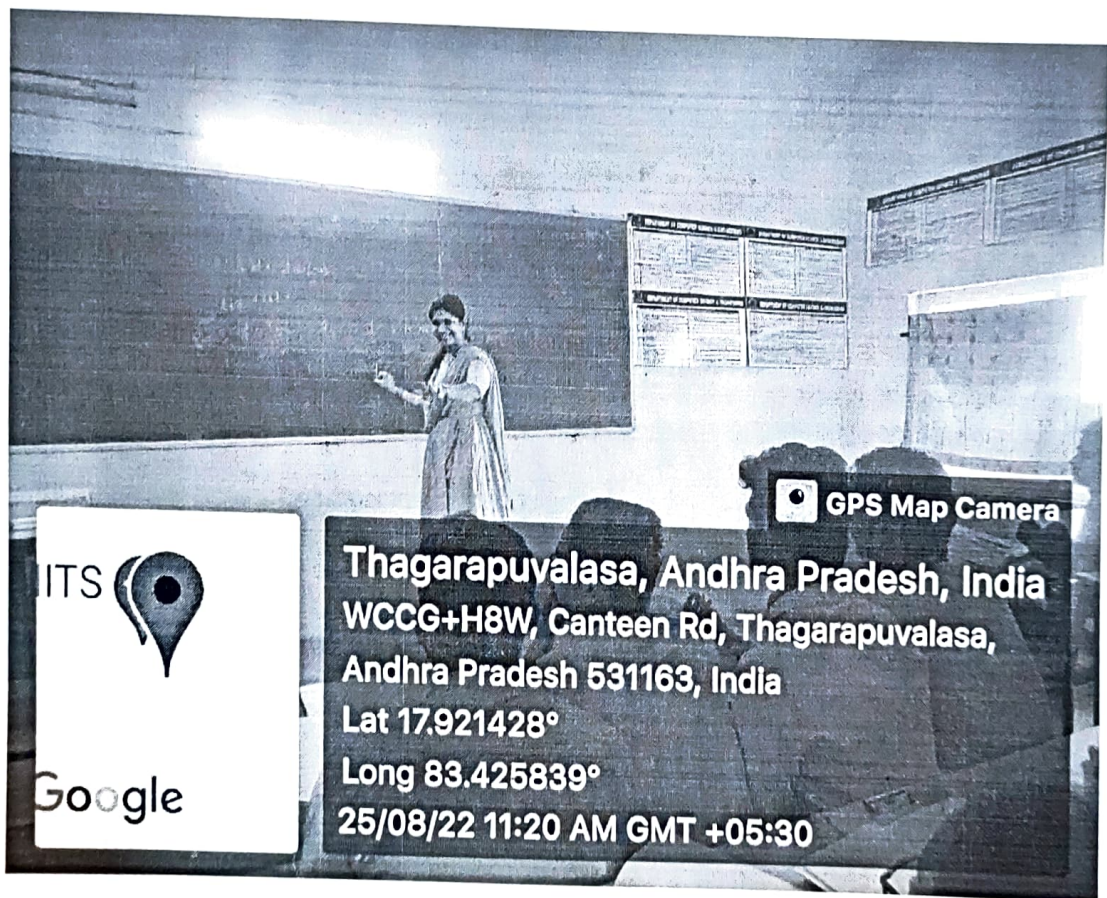
Like stacks, Queues can also be represented in an array: In this representation, the Queue is implemented using the array. Variables used in this case are

Queue: the name of the array storing queue elements.

Front: the index where the first element is stored in the array representing the queue.

Rear: the index where the last element is stored in an array representing the queue.

Photos:



*Dulip*  
Head of the Department of  
Computer Science & Engineering  
Anil Neerukonda Institute of  
Technology & Sciences  
Sangivalasa, Visakhapatnam Dist



GPS Map Camera

**Thagarapuvalasa, Andhra Pradesh, India**  
**WCCG+H8W, Canteen Rd, Thagarapuvalasa,**  
**Andhra Pradesh 531163, India**  
**Lat 17.921428°**  
**Long 83.425819°**  
**25/08/22 11:18 AM GMT +05:30**

*Anil Kumar*  
Head of the Department of  
Computer Science & Engineering  
Anil Neerlokaj Institute of  
Technology & Sciences  
Sangivalasa, Visakhapatnam Dist

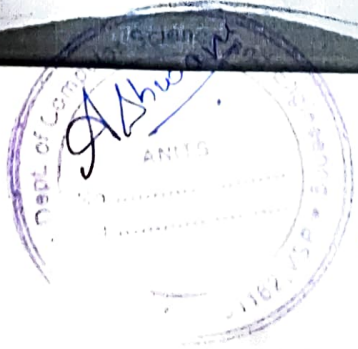
Attendance:

25/8/22

ANIL NEERUKONDA INSTITUTE OF TECHNOLOGY AND SCIENCES (A)  
DEPARTMENT OF CSE  
3-4 B. Tech CSE - C SEM - I

GATE Classes

S.No	Roll No	Name of the student	Signature
1	320126510129	AKANKSHA VEERAMALLA	Akanksha
2	320126510130	ALLA CHANDU	Chandu
3	320126510131	AMIREDDI SIVAKUMAR	Siva Kumar
4	320126510132	ANANTA RAGHAVENDRA GOKUL ALURI	Gokul
5	320126510133	ARASADA SAI ANUSHA	Anusha
6	320126510134	AVALA GUNASRI	Gunasi
7	320126510135	BALAMURALI BESETTY	Besetty
8	320126510136	BALI HARSHITHA	Harshitha
9	320126510137	BANDI GOWTHAM	Gowtham
10	320126510138	BARRI HEMA RAJU	Hemara
11	320126510139	BONTHALAKOTI KRISHNA SAI KARTHIK CHANDRA	Karthik
12	320126510140	CHANDANA CHARITHA PEDDINTI	Charitha
13	320126510141	CHINTALA MAHIMA RISHITA	Rishita
14	320126510142	DALAI JAYANTHI KUMAR	Jayanthi
15	320126510143	DODDI LALITHA	Lalitha
16	320126510144	GNANODHAY RANDHI	Gnanodhay
17	320126510145	GONNABATHULA DEV HARSHI	Devharshi
18	320126510146	GUNTUKU SAI CHARAN	Charan
19	320126510147	GURUBELLI VENKATA SURYA SAI	Surya Sai
20	320126510148	ITHA SAI DEEPTI	Deepti
21	320126510149	ITRAJU DILLESWARA RAO	Dilleshwar
22	320126510150	K NAVEEN KUMAR	Naveen Kumar
23	320126510151	KANDUKURI VINEELA	Vineela
24	320126510152	KETHANAPALLI SHANMUKHA SRINIVAS	Shanmukha
25	320126510153	KINTALI SRI VARSHA	Varsha
26	320126510154	KOPPALA KUSUMA PRIYA	Kusuma Priya
27	320126510155	KOVIRI MADHU BABU	Madhu
28	320126510156	KUMMARI KIRANKUMAR	Kirankumar
29	320126510157	MADDI SAI SANTOSHI	Santisoshi
30	320126510158	MANYAM ASRITHA	Asritha
31	320126510159	MEELA DHEERAJ	Dheeraj
32	320126510160	MEESALA SAI MADHUKAR	Madhukar
33	320126510161	MOGASALA SANJEEVARAYA ADITYA VARDHAN	Aditya Vardhan
34	320126510162	MYLABATHULA BIHARATHI	Biharathi
35	320126510163	NAGARAJU AAKASHI	Aakashi
36	320126510164	NISHITHA GALLA	Nishitha
37	320126510165	PAIDI AKILESWAR	Akileswar
38	320126510166	PEKETI SHANMUK SAI MANOJ	Shanmuk Sai Manoj
39	320126510167	PENTAPALLI DEEPA PRIYANKA	Deepa Priyanka
40	320126510168	PINNINTI GEETHA RANI	Geetha Rani
41	320126510169	PONDALA ANIL CHANDRA	Anil Chandra
42	320126510170	PUDI INDRANI	Indrani
43	320126510171	RAPAKA HEPIZIBAH	Hepizibah
44	320126510172	REGI LAVANYAKUMAR	Lavanyakumar
45	320126510173	SABHAVARAPU SAI KISHORE	Saibhavapuru Sai Kishore



*Srikanth*  
Head of the Department of  
Computer Science & Engineering  
Anil Neerukonda Institute of  
Technology & Sciences  
Sangivalasa, Visakhapatnam Dist

25/10/22

ANIL NEERUKONDA INSTITUTE OF TECHNOLOGY AND SCIENCES (A)  
DEPARTMENT OF CSE  
3-4 B. Tech CSE - A SEM - I

GATE Class


S.No	Roll No	Induction Programme Name of the student	Signature
1	320126510001	ADAPA SASHANK	[Signature]
2	320126510002	ADAPAKA NEELIMA	[Signature]
3	320126510003	AGNIHOTRAM SRI VENKATA SAKIRAN CHAITANYA	[Signature]
4	320126510004	ANNEPU HEMANTH KUMAR	[Signature]
5	320126510005	BANDARU VENKATA JAYARAM	[Signature]
6	320126510006	BETHA AJAY KUMAR	[Signature]
7	320126510007	BETHA AKASH	[Signature]
8	320126510008	BITRA VENKAT SAI KIRAN	[Signature]
9	320126510009	BODDU THANMAI	[Signature]
10	320126510010	BOOSI ARAVIND	[Signature]
11	320126510011	BUDHA GOWTHAM	[Signature]
12	320126510012	BULASARA LIKHIT SRI SAI	[Signature]
13	320126510013	CHIGATI SATWIK	[Signature]
14	320126510014	CHINNAPARAPU AJAY ESWAR	[Signature]
15	320126510015	CHINTA DEEPTHI	[Signature]
16	320126510016	DEVIKA KONDAPALLI	[Signature]
17	320126510017	G SUNIL	[Signature]
18	320126510018	GANDI SARAN KUMAR	[Signature]
19	320126510019	GANGADEVI LAKSHMI PRAVALLIKA	[Signature]
20	320126510020	GELLA CHAITANYA VENKATA SAI	[Signature]
21	320126510021	GOPSETTI ANIL KUMAR	[Signature]
22	320126510022	GOSALA SURYA VARDHAN REDDY	[Signature]
23	320126510023	GUDIWADA NAGA LAKSHMI NAVEEN	[Signature]
24	320126510024	GUDLA TEJESH	[Signature]
25	320126510025	GUNDU ABHINAYA	[Signature]
26	320126510026	INTI AAKASH	[Signature]
27	320126510027	JARAJANA HARSHIT	[Signature]
28	320126510028	KARICHETI SIVA KALYAN	[Signature]
29	320126510029	KASAMSETTI MANOU SAMPATH	[Signature]
30	320126510030	KASINA CHANDI NAGA PAVAN RAJESH	[Signature]
31	320126510031	KASIREDDI SAI SHRUTHI	[Signature]
32	320126510032	KOLLI ANANTHA RAO	[Signature]
33	320126510033	KOMPILLI RAJ KUMAR	[Signature]
34	320126510034	KONDAVETI ANUPAMA	[Signature]
35	320126510035	KOTA CHANDINI MANOZNA	[Signature]
36	320126510036	KUMBHA NARENDRA KUMAR	[Signature]
37	320126510037	KURAKULA ROHIT	[Signature]
38	320126510038	LALITH KIRAN PEERUKATTLA	[Signature]
39	320126510039	LODAGALA HEMA SAI	[Signature]
40	320126510040	M MOHIDEEN FATHIMA NILOFAR	[Signature]
41	320126510041	MAHATO USHA	[Signature]
42	320126510042	MARADA JOSHIITH	[Signature]
43	320126510043	MATTA BHASHITHA	[Signature]
44	320126510044	METTU AKHILESH	[Signature]



*[Handwritten Signature]*  
 Head of the Department of  
 Computer Science & Engineering  
 Anil Neerukonda Institute of  
 Technology & Sciences  
 Sangli

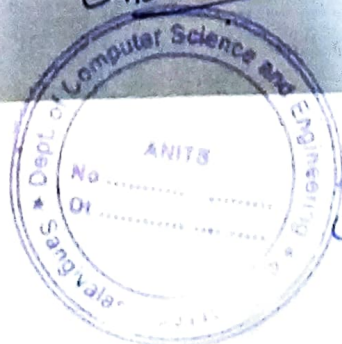
S.No	Roll No	Name of the student	Signature
46	320126510174	SAGIRAJU VENKATA SIVA LOHIT VARMA	Sagiraju
47	320126510175	SALUGU SURYA KRISHNA KARTHIK	Salugu
48	320126510176	SANJEEVU SANDEEP KUMAR	Sanjeevu
49	320126510177	SHAIK FOUZIYA	Shaike
50	320126510178	SHIV VARMA PUSAPATI	Shiv Varma
51	320126510179	SINGAMPALLI SRIHARSHA	Singampalli
52	320126510180	SURAKASI BHARAT	Surakasi
53	320126510181	THATIPUDI DEEPIKA	Thatipudi
54	320126510182	THOLETI KUSUMA SRI	Tholeti
55	320126510183	THOTA BALAJI	Thota
56	320126510184	VADADA SHILPA SHARMILA	Vadada
57	320126510185	VALLEPU RAMA HARESHKIRAN	Vallepu
58	320126510186	VARADA VASAVI	Varada
59	320126510187	VEDURUPAKALA JYOTHSNA	Vedurupakala
60	320126510188	VISINIGIRI UDAY KIRAN	Visinigiri
61	320126510189	VOONA PERIN KRISHNA	Voona
62	320126510190	YADLA HARSHAVARDHAN SAI	Yadla
63	320126510191	YENUGUTALA SURESH	Yenugutala
64	321126510L13	PEDDINTI NANDINI	Peddinti
65	321126510L14	MALLETHANSI	Mallethansi
66	321126510L15	SABBISSETTY MADHAVI	Sabbisetty
67	321126510L16	KONATHALA VEDHA ADI VIDITHA	Konathala
68	321126510L17	VOLATI SAIKIRAN	Volati
69	321126510L18	ANDAVARAPU CHANDU	Andavarapu
70	321126510L19	PAVAN KUMAR PONNAGANTI	Pavan Kumar
71	321126510L20	K VENKATA ARAVINDA REDDY	K Venkata Aravinda



  
 Head of the Department of  
 Computer Science & Engineering  
 Anil Neeraj Institute of  
 Technology & Sciences  
 Sangivalasa, Visakhapatnam Dist

S.No	Roll No	Name of the student	Signature
45	320126510045	MOLLETI YERNI TRIBHUVAN	M. Y. Kumar
46	320126510046	MUMMIDIVARAPU VENKATESH	M. Venkatesh
47	320126510048	NANMI ADITYA	N. Aditya
48	320126510049	PALLE TARUN KUMAR	P. Tarun Kumar
49	320126510050	PANANGIPALLI KARTHIKEYA	K. K.
50	320126510051	PANDA ANJALI	A. P.
51	320126510052	PASUPUREDDI HARITHA	H. P.
52	320126510053	PATTAMATTA NAGA HIMA VARSHA	N. H. V.
53	320126510054	PATTIKA RAJESH	P. R.
54	320126510055	PUNUGOTI CHANDRA SEKHAR	C. S.
55	320126510056	RAGHAVENDRA ANURAG SETAMRAJU	R. A. S.
56	320126510057	RUSHIKESH MUKUND PATMASE	R. M. P.
57	320126510058	SALA VENKATA NAGA SAI CHARAN	S. V. N. S. C.
58	320126510059	SANAPALA NIKHILESH	S. N.
59	320126510060	SEEPANA MITHUN	S. M.
60	320126510061	SHAIK BASHIR AHAMED	S. B. A.
61	320126510062	VANAM LALITHA SHREYA	V. L. S.
62	320126510063	VEMA VENKATA YESWANTH GUPITHA	V. V. Y. G.
63	320126510064	VYSYARAJU RAMAN RAJ	V. R. R.
64	321126510L01	KARRI SAI KUMAR	K. S. K.
65	321126510L02	MAJI HEMANTH REDDY	M. H. R.
66	321126510L03	KURMAPU DHARANEESWAR	K. D.
67	321126510L04	DONI TWINKLE	D. T.
68	321126510L05	AMARA NAGENDRA	A. N.
69	321126510L06	SABBAVARAPU USHA SRI	S. U. S.

Ashwani



Anil Neerukoti Institute of  
 Computer Science & Engineering  
 Technology & Sciences  
 Sangivalasa, Visakhapatnam Dist.

25/8/22

## ANIL NEERUKONDA INSTITUTE OF TECHNOLOGY AND SCIENCES (A)

DEPARTMENT OF CSE

GATE Classes

## 3-4 B. Tech CSE - A SEM - I

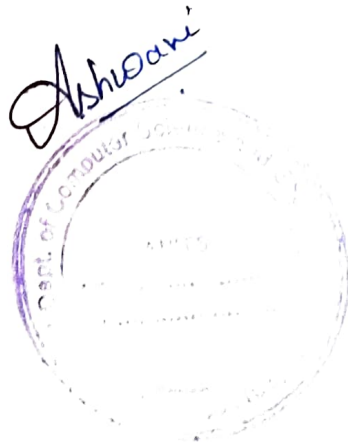
## Induction Programme

S.No	Roll No	Name of the student	Signature
1	320126510001	ADAPA SASHANK	Sashank
2	320126510002	ADAPAKA NEELIMA	A. Neelima
3	320126510003	AGNIHOTRAM SRI VENKATA SAIKIRAN CHAITANYA	Chaitanya - A
4	320126510004	ANNEPU HEMANTH KUMAR	Hemant
5	320126510005	BANDARU VENKATA JAYARAM	JAYARAM B
6	320126510006	BETHA AJAY KUMAR	Ajay Kumar
7	320126510007	BETHA AKASH	B. Akash
8	320126510008	BITRA VENKAT SAI KIRAN	B. Kiran
9	320126510009	BODDU THANMAI	Thanmai
10	320126510010	BOOSI ARAVIND	Aravind - B.
11	320126510011	BUDHA GOWTHAM	Gowtham
12	320126510012	BULASARA LIKHIT SRI SAI	Likhith Sri Sai
13	320126510013	CHIGATI SATWIKA	Ch. Satwika
14	320126510014	CHINNAPARAPU AJAY ESWAR	Ajay Eswar
15	320126510015	CHINTA DEEPTHI	Deepthi
16	320126510016	DEVIKA KONDAPALLI	Devika
17	320126510017	G SUNIL	Sunil
18	320126510018	GANDI SARAN KUMAR	Saran Kumar
19	320126510019	GANGADEVI LAKSHMI PRAVALLIKA	
20	320126510020	GELLA CHAITANYA VENKATA SAI	Chaitanya
21	320126510021	GOPISETTI ANIL KUMAR	Anil Kumar
22	320126510022	GOSALA SURYA VARDHAN REDDY	Surya
23	320126510023	GUDIWADA NAGA LAKSHMI NAVEEN	Naveen
24	320126510024	GUDLA TEJESH	Tejesh
25	320126510025	GUNDU ABHINAYA	Abhinaya
26	320126510026	INTI AAKASH	Aakash
27	320126510027	JARAJANA HARSHIT	
28	320126510028	KARICHETI SIVA KALYAN	Sivakalyan
29	320126510029	KASAMSETTI MANOJ SAMPATH	Manoj
30	320126510030	KASINA CHANDI NAGA PAVAN RAJESH	Pavan
31	320126510031	KASIREDDI SAI SHRUTHI	Shruthi
32	320126510032	KOLLI ANANTHA RAO	Anantha Rao
33	320126510033	KOMPILLI RAJ KUMAR	Raj Kumar
34	320126510034	KONDAVETI ANUPAMA	Anupama
35	320126510035	KOTA CHANDINI MANOZNA	Chandini
36	320126510036	KUMBHA NARENDRA KUMAR	Narendra
37	320126510037	KURAKULA ROHIT	Rohit
38	320126510038	LALITH KIRAN PEERUKATTLA	Kiran
39	320126510039	LODAGALA HEMA SAI	Hema Sai
40	320126510040	M MOHIDEEN FATHIMA NILOFAR	Fathima Nilofar
41	320126510041	MAHATO USHA	Usha
42	320126510042	MARADA JOSHITH	Joshith
43	320126510043	MATTA BHASHITHA	Bhashitha
44	320126510044	METTU AKHILESH	Akhilesh



S.No	Roll No	Name of the student	Signature
45	320126510045	MOLLETI YERNI TRIBHUVAN	M. Tribhuvan
46	320126510046	MUMMIDIVARAPU VENKATESH	M. Venkatesh
47	320126510048	NAMMI ADITYA	N. Aditya
48	320126510049	PALLE TARUN KUMAR	P. Tarun Kumar
49	320126510050	PANANGIPALLI KARTHIKEYA	Karthikeya
50	320126510051	PANDA ANJALI	Anjali
51	320126510052	PASUPUREDDI HARITHA	Haritha
52	320126510053	PATTAMATTA NAGA HIMA VARSHA	Hima Varsha
53	320126510054	PATTIKA RAJESH	Rajesh
54	320126510055	PUNUGOTI CHANDRA SEKHAH	Chandrasekhar
55	320126510056	RAGHAVENDRA ANURAG SETAMRAJU	Anurag
56	320126510057	RUSHIKESH MUKUND PATMASE	Rushikesh
57	320126510058	SALA VENKATA NAGA SAI CHARAN	Sai Charan
58	320126510059	SANAPALA NIKHILESH	Nikhilesh
59	320126510060	SEEPANA MITHUN	Mithun
60	320126510061	SHAIK BASHEER AHAMED	Basheer
61	320126510062	VANAM LALITHA SHREYA	Lalitha
62	320126510063	VEMA VENKATA YESWANTH GUPTHA	Yeswanth
63	320126510064	VYSYARAJU RAMAN RAJ	Raman Raj
64	321126510L01	KARRI SAI KUMAR	Sai Kumar
65	321126510L02	MAJJI HEMANTH REDDY	Hemant
66	321126510L03	KURMAPU DHARANEESWAR	Dhara
67	321126510L04	DONI TWINKLE	Twinkle
68	321126510L05	AMARA NAGENDRA	Nagendra
69	321126510L06	SABBAVARAPU USHA SRI	Usha Sri

*S. S. Prasad*  
Head of the Department of  
Computer Science & Engineering  
Anil Neerukonda Institute of  
Technology & Sciences  
Sangivalasa, Visakhapatnam Dist



25/8/22

## ANIL NEERUKONDA INSTITUTE OF TECHNOLOGY AND SCIENCES (A)

## DEPARTMENT OF CSE

GATE Classes

## 2-4 B. Tech CSE - C SEM - I

S.No	Roll No	Name of the student	Signature
1	320126510129	AKANKSHA VEERAMALLA	Akanksha
2	320126510130	ALLA CHANDU	chandu
3	320126510131	AMIREDDI SIVAKUMAR	Siva kuma
4	320126510132	ANANTA RAGHAVENDRA GOKUL ALURI	Gokul
5	320126510133	ARASADA SAI ANUSHA	Anusha
6	320126510134	AVALA GUNASRI	Gunasri
7	320126510135	BALAMURALI BESETTY	Balesri
8	320126510136	BALI HARSHITHA	Harshitha
9	320126510137	BANDI GOWTHAM	Gowtham
10	320126510138	BARRI HEMA RAJU	Hemara
11	320126510139	BONTHALAKOTI KRISHNA SAI KARTHIK CHANDRA	Karthik
12	320126510140	CHANDANA CHARITHA PEDDINTI	Charitha
13	320126510141	CHINTALA MAHIMA RISHITA	Rishitha
14	320126510142	DALAI JAYANTH KUMAR	Jayanth
15	320126510143	DODDI LALITHA	Lalitha
16	320126510144	GNANODHAY RANDHI	Gnanodhay
17	320126510145	GONNABATHULA DEV HARSH	Dev
18	320126510146	GUNTUKU SAI CHARAN	Charan
19	320126510147	GURUBELLI VENKATA SURYA SAI	Surya
20	320126510148	ITHA SAI DEEPTI	Deepti
21	320126510149	ITRAJU DILLESWARA RAO	Dilleshwar
22	320126510150	K NAVEEN KUMAR	Naveen kumar
23	320126510151	KANDUKURI VINEELA	
24	320126510152	KETHANAPALLI SHANMUKHA SRINIVAS	Shanmukha
25	320126510153	KINTALI SRI VARSHA	Varsha
26	320126510154	KOPPALA KUSUMA PRIYA	Kusuma Priya
27	320126510155	KOVIRI MADHU BABU	Madhu
28	320126510156	KUMMARI KIRANKUMAR	Kiran Kumar
29	320126510157	MADDI SAI SANTOSH	Santosh
30	320126510158	MANYAM ASRITHA	Asritha
31	320126510159	MEELA DHEERAJ	
32	320126510160	MEESALA SAI MADHUKAR	Madhukar
33	320126510161	MOGASALA SANJEEVARAYA ADITYA VARDHAN	
34	320126510162	MYLABATHULA BHARATH	Bharath
35	320126510163	NAGARAJU AAKASHI	Aakash Arjun
36	320126510164	NISHITHA GALLA	Nishitha
37	320126510165	PAIDI AKILESWAR	Akili
38	320126510166	PEKETI SHANMUK SAI MANOJ	Shanmukh saji
39	320126510167	PENTAPALLI DEEPA PRIYANKA	
40	320126510168	PINNINTI GEETHA RANI	Geetha
41	320126510169	PONDALA ANIL CHANDRA	Anil chandra
42	320126510170	PUDI INDRANI	Indrani
43	320126510171	RAPAKA HEPIZIBAH	Hepzibah
44	320126510172	REGI LAVANYAKUMAR	Lavanya Kumar
45	320126510173	SABBAVARAPU SAI KISHORE	Kishore

Balu prasad

S.No	Roll No	Name of the student	Signature
46	320126510174	SAGIRAJU VENKATA SIVA LOHIT VARMA	Sagira
47	320126510175	SALUGU SURYA KRISHNA KARTHIK	Salugu
48	320126510176	SANJEEVU SANDEEP KUMAR	Sanjeevu
49	320126510177	SHAIK FOUZIYA	Shaik
50	320126510178	SHIV VARMA PUSAPATI	Shiv
51	320126510179	SINGAMPALLI SRIHARSHA	Singampalli
52	320126510180	SURAKASI BHARAT	Surakasi
53	320126510181	THATIPUDI DEEPIKA	Thatipudi
54	320126510182	THOLETI KUSUMA SRI	Tholeti
55	320126510183	THOTA BALAJI	
56	320126510184	VADADA SHILPA SHARMILA	
57	320126510185	VALLEPU RAMA HARESHKIRAN	Vallepu
58	320126510186	VARADA VASAVI	Varada
59	320126510187	VEDURUPAKALA JYOTHSNA	Vedurupakala
60	320126510188	VISINIGIRI UDAY KIRAN	Visinigiri
61	320126510189	VOONA PERIN KRISHNA	Voona
62	320126510190	YADLA HARSHAVARDHAN SAI	Yadla
63	320126510191	YENUGUTALA SURESH	Yenugutala
64	321126510L13	PEDDINTI NANDINI	
65	321126510L14	MALLETI JHANSI	Malleti
66	321126510L15	SABBISSETTY MADHAVI	Sabbisetty
67	321126510L16	KONATHALA VEDHA ADI VIDITHA	Konathala
68	321126510L17	VOLATI SAIKIRAN	Volati
69	321126510L18	ANDAVARAPU CHANDU	Andavarapu
70	321126510L19	PAVAN KUMAR PONNAGANTI	Pavan
71	321126510L20	K VENKATA ARAVINDA REDDY	K Venkata

*Anil Neerula*

Head of the Department of  
Computer Science & Engineering  
Anil Neerula Institute of  
Technology & Sciences  
Sangivalasa, Visakhapatnam Dist





**Anil Neerukonda Institute of Technology & Sciences  
(Autonomous)**

(Approved to AU, Approved by AICTE & Accredited by NBA (ECE/EE/CSE/IT & Mech) & NAAC)  
Narayanpet, 511 162, Dharmavaram Mandal, Vengal Rao District  
Phone: 0844 22 908 30 40 / Fax: 226195  
Website: www.anits.edu.in email: principal@anits.edu.in

Date: 26-08-2022

Topic: Programming

Faculty: Dr.V. Sangeetha, Asst Professor.

On 26-08-2022, V.Sangeetha addressed the students on Data Structures related to Gate Examinations to all the students. It was an interactive session with the students where the students understood all the topics clearly.

Abstract:

- C Programming,
- Java Programming

C Programming:

C is a general-purpose programming language that is extremely popular, simple, and flexible to use. It is a structured programming language that is machine-independent and extensively used to write various applications, Operating Systems like Windows, and many other complex programs like Oracle database, Git, Python interpreter, and more.

It is said that 'C' is a god's programming language. One can say, C is a base for programming. If you know 'C,' you can easily grasp the knowledge of the other programming languages that use the concept of 'C'

It is essential to have a background in computer memory mechanisms because it is an important aspect when dealing with the C programming language.

**C Basic Commands**

Following are the basic commands in C programming language:

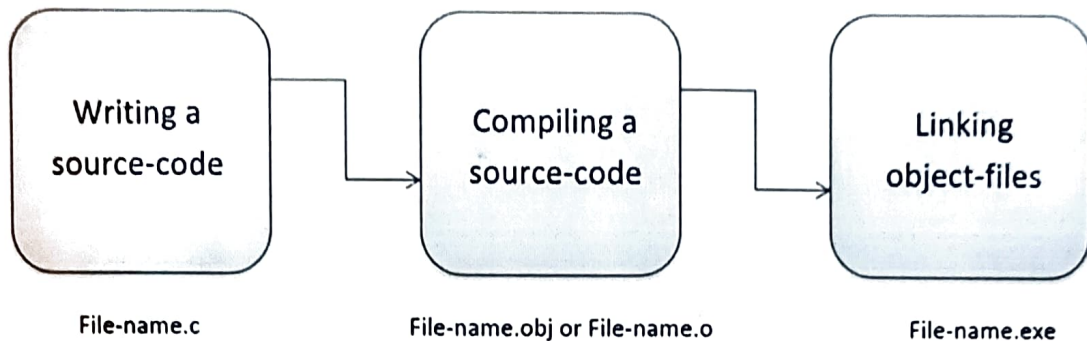
C Basic commands	Explanation
<code>#include &lt;stdio.h&gt;</code>	This command includes standard input output header file(stdio.h) from the C library before compiling a C program

<code>int main()</code>	It is the main function from where C program execution begins.
<code>{</code>	Indicates the beginning of the main function.
<code>/*_some_comments_*/</code>	Whatever written inside this command <code>/* */</code> inside a C program, it will not be considered for compilation and execution.
<code>printf("Hello_World! ");</code>	This command prints the output on the screen.
<code>getch();</code>	This command is used for any character input from keyboard.
<code>return 0;</code>	This command is used to terminate a C program (main function) and it returns 0.
<code>}</code>	It is used to indicate the end of the main function.

#### Applications:

1. 'C' language is widely used in embedded systems.
2. It is used for developing system applications.
3. It is widely used for developing desktop applications.
4. Most of the applications by Adobe are developed using 'C' programming language.
5. It is used for developing browsers and their extensions. Google's Chromium is built using 'C' programming language.
6. It is used to develop databases. MySQL is the most popular database software which is built using 'C'.
7. It is used in developing an operating system. Operating systems such as Apple's OS X, Microsoft's Windows, and Symbian are developed using 'C' language. It is used for developing desktop as well as mobile phone's operating system.
8. It is used for compiler production.
9. It is widely used in IOT applications.

C is a compiled language. A compiler is a special tool that compiles the program and converts it into the object file which is machine readable. After the compilation process, the linker will combine different object files and creates a single executable file to run the program. The following diagram shows the execution of a 'C' program



Nowadays, various compilers are available online, and you can use any of those compilers. The functionality will never differ and most of the compilers will provide the features required to execute both 'C' and 'C++' programs.

### Java Programming Language:

**Object-Oriented Programming System (OOPs)** is a programming concept that works on the principles of abstraction, encapsulation, inheritance, and polymorphism. It allows users to create objects they want and create methods to handle those objects. The basic concept of OOPs is to create objects, re-use them throughout the program, and manipulate these objects to get results.

OOP meaning "Object Oriented Programming" is a popularly known and widely used concept in modern programming languages like Java.

### OOPs Concepts in Java with Examples

The following are general OOPs concepts in Java:

- 1) Class

The class is one of the Basic concepts of OOPs which is a group of similar entities. It is only a logical component and not the physical entity. Lets understand this one of the OOPs Concepts with example, if you had a class called "Expensive Cars" it could have objects like Mercedes, BMW, Toyota, etc. Its properties(data) can be price or speed of these cars. While the methods may be performed with these cars are driving, reverse, braking etc.

## 2) Object

An object can be defined as an instance of a class, and there can be multiple instances of a class in a program. An Object is one of the Java OOPs concepts which contains both the data and the function, which operates on the data. For example – chair, bike, marker, pen, table, car, etc.

## 3) Inheritance

Inheritance is one of the Basic Concepts of OOPs in which one object acquires the properties and behaviors of the parent object. It's creating a parent-child relationship between two classes. It offers robust and natural mechanism for organizing and structure of any software.

## 4) Polymorphism

Polymorphism refers to one of the OOPs concepts in Java which is the ability of a variable, object or function to take on multiple forms. For example, in English, the verb run has a different meaning if you use it with a laptop, a foot race, and business. Here, we understand the meaning of run based on the other words used along with it. The same also applied to Polymorphism.

## 5) Abstraction

Abstraction is one of the OOP Concepts in Java which is an act of representing essential features without including background details. It is a technique of creating a new data type that is suited for a specific application. Lets understand this one of the OOPs Concepts with example, while driving a car, you do not have to be concerned with its internal working. Here you just need to concern about parts like steering wheel, Gears, accelerator, etc.

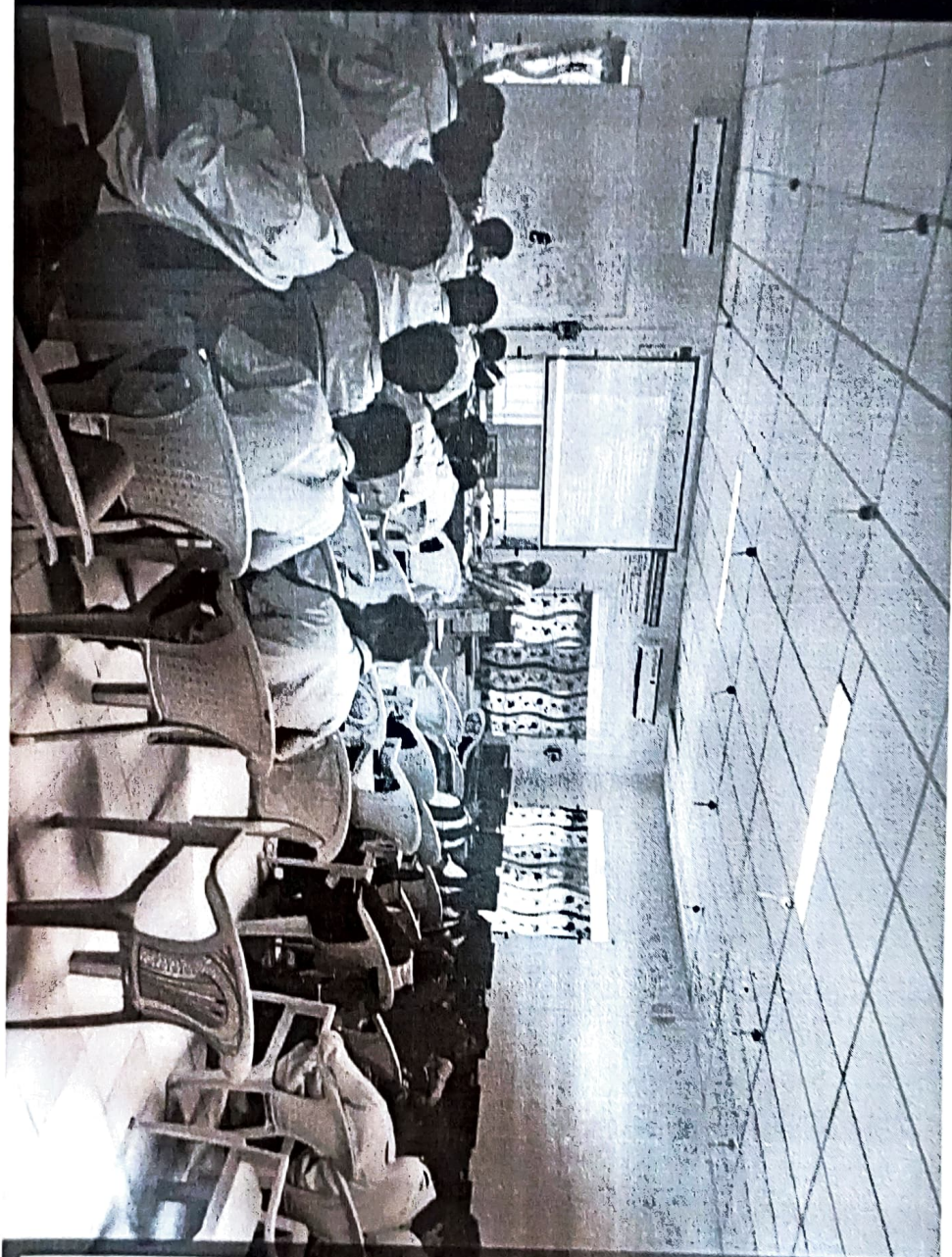
## 6) Encapsulation

Encapsulation is one of the best Java OOPs concepts of wrapping the data and code. In this OOPs concept, the variables of a class are always hidden from other classes. It can only be accessed using the methods of their current class. For example – in school, a student cannot exist without a class.

## 7) Association

Association is a relationship between two objects. It is one of the OOP Concepts in Java which defines the diversity between objects. In this OOP concept, all objects have their separate lifecycle, and there is no owner. For example, many students can associate with one teacher while one student can also associate with multiple teachers.

## 8) Aggregation



*Baru - pusey*



WCCG+H8W, Canteen Rd,  
Thagarapuvalasa, Andhra  
Pradesh 531163, India  
17 Sep 2022 01:54 pm

overcast  
clouds  
34.0 °C





WCCG+H8W, Canteen Rd,  
Thagarapavalasa, Andhra  
Pradesh 531163, India  
overcast clouds  
34.0 °C  
17 Sep 2022 01:54 pm

*S. S. Jureel*  
Head of the Department of  
Computer Science & Engineering  
Anil Neerukonda Institute of  
Technology & Sciences  
Sangivalasa, Visakhapatnam Dist

26/8/22

## ANIL NEERUKONDA INSTITUTE OF TECHNOLOGY AND SCIENCES (A)

## DEPARTMENT OF CSE

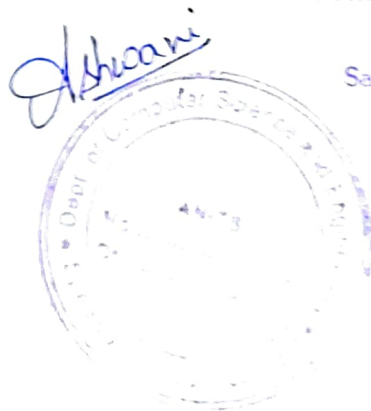
GATE Classes

## 2-4 B. Tech CSE - C SEM - I

S.No	Roll No	Name of the student	Signature
1	320126510129	AKANKSHA VEERAMALLA	v Akanksha
2	320126510130	ALLA CHANDU	Chandu
3	320126510131	AMIREDDI SIVAKUMAR	A Sivakumar
4	320126510132	ANANTA RAGHAVENDRA GOKUL ALURI	Gokul
5	320126510133	ARASADA SAI ANUSHA	Anusha
6	320126510134	AVALA GUNASRI	GUNA SRI
7	320126510135	BALAMURALI BESETTY	Murati
8	320126510136	BALI HARSHITHA	Harshitha
9	320126510137	BANDI GOWTHAM	
10	320126510138	BARRI HEMA RAJU	Hemastaja
11	320126510139	BONTHALAKOTI KRISHNA SAI KARTHIC CHANDRA	B.Chandrag
12	320126510140	CHANDANA CHARITHA PEDDINTI	Chandane
13	320126510141	CHINTALA MAHIMA RISHITA	
14	320126510142	DALAI JAYANTH KUMAR	Jayanth
15	320126510143	DODDI LALITHA	Lalitha
16	320126510144	GNANODHAY RANDHI	Gnanodhay
17	320126510145	GONNABATHULA DEV HARSH	Dev Harsh
18	320126510146	GUNTUKU SAI CHARAN	Charan
19	320126510147	GURUBELLI VENKATA SURYA SAI	Surya
20	320126510148	ITHA SAI DEEPTI	deepti
21	320126510149	ITRAJU DILLESWARA RAO	Dilli
22	320126510150	K NAVEEN KUMAR	Vineela
23	320126510151	KANDUKURI VINEELA	Naveen
24	320126510152	KETHANAPALLI SHANMUKHA SRINIVAS	k.s.s.srinivas
25	320126510153	KINTALI SRI VARSHA	ponya
26	320126510154	KOPPALA KUSUMA PRIYA	Madu Babu
27	320126510155	KOVIRI MADHU BABU	M. Santhosh
28	320126510156	KUMMARI KIRANKUMAR	
29	320126510157	MADDI SAI SANTOSH	Madhu
30	320126510158	MANYAM ASRITHA	Asritha
31	320126510159	MEELA DHEERAJ	Dheeraj
32	320126510160	MEESALA SAI MADHUKAR	Madhukar
33	320126510161	MOGASALA SANJEEVARAYA ADITYA VARDHAN	
34	320126510162	MYLABATHULA BHARATH	N. A Akash
35	320126510163	NAGARAJU AAKASH	
36	320126510164	NISHITHA GALLA	G. Nishitha
37	320126510165	PAIDI AKILESWAR	Sai Manoj
38	320126510166	PEKETI SHANMUK SAI MANOJ	
39	320126510167	PENTAPALLI DEEPA PRIYANKA	Geetha Rani
40	320126510168	PINNINTI GEETHA RANI	Deepa
41	320126510169	PONDALA ANIL CHANDRA	Anil
42	320126510170	PUDI INDRANI	
43	320126510171	RAPAKA HEPHIZIBAH	Hephzibah
44	320126510172	REGI LAVANYAKUMAR	
45	320126510173	SABBAVARAPU SAI KISHORE	Sai Kishore

S.No	Roll No	Name of the student	Signature
46	320126510174	SAGIRAJU VENKATA SIVA LOHIT VARMA	S S S Lohit
47	320126510175	SALUGU SURYA KRISHNA KARTHIK	S S K Karthik
48	320126510176	SANJEEVU SANDEEP KUMAR	Sandeep
49	320126510177	SHAIK FOUZIYA	Fouziya
50	320126510178	SHIV VARMA PUSAPATI	Shiv Varma
51	320126510179	SINGAMPALLI SRIHARSHA	
52	320126510180	SURAKASI BHARAT	Bharat
53	320126510181	THATIPUDI DEEPIKA	Deepika
54	320126510182	THOLETI KUSUMA SRI	
55	320126510183	THOTA BALAJI	Balaji
56	320126510184	VADADA SHILPA SHARMILA	
57	320126510185	VALLEPU RAMA HARESHKIRAN	Hareshkiran
58	320126510186	VARADA VASAVI	V. Vasavi
59	320126510187	VEDURUPAKALA JYOTHSNA	V. Jyothsna
60	320126510188	VISINIGIRI UDAY KIRAN	
61	320126510189	VOONA PERIN KRISHNA	perin krishna
62	320126510190	YADLA HARSHAVARDHAN SAI	harshardh.
63	320126510191	YENUGUTALA SURESH	
64	321126510L13	PEDDINTI NANDINI	Nandini
65	321126510L14	MALLETI JHANSI	Jhansi
66	321126510L15	SABBISSETTY MADHAVI	madhavi
67	321126510L16	KONATHALA VEDHA ADI VIDITHA	viditha
68	321126510L17	VOLATI SAIKIRAN	V. Saikiran
69	321126510L18	ANDAVARAPU CHANDU	chandu
70	321126510L19	PAVAN KUMAR PONNAGANTI	pavan kumar
71	321126510L20	K VENKATA ARAVINDA REDDY	Aravind

*Bulpujef*  
Head of the Department of  
Computer Science & Engineering  
Andhra Pradesh State Institute of  
Technical Education  
Sangwalasa, Visakhapatnam Dist



26/8/22

ANIL NEERUKONDA INSTITUTE OF TECHNOLOGY AND SCIENCES (A)

DEPARTMENT OF CSE

24 B. Tech CSE - B SEM - II

GATE CLASSES

S.No	Roll No	Name of the student	Signature
1	320126510065	ALAJANGI LIKHITHA	Likhitha
2	320126510066	APPALABATHULA SATYA LOKESH	Satya Lokesh
3	320126510067	ARIPAKA VINAY	Vinay
4	320126510068	B UHA MADHURI	B. Uha Madhuri
5	320126510069	BAJJANGI CHAITANYA NANDA KUMAR	Chaitanya Nanda Kumar
6	320126510070	BODA VASU DEVA RAO	Vasu Deva Rao
7	320126510071	BONULA GOWTHAMI	Gowthami
8	320126510072	BOTTA SURYA SYAM SAI SWETHAK	Swethak
9	320126510073	BULUSU SRI LAKSHMI MANASA	Manasa
10	320126510074	CHAPPA ANEESH	Aneesh
11	320126510075	CHIDAM VINAY MANIKANTA KARTHIK	Karthik
12	320126510076	CHIDARLLA VIJAY KUMAR	Vijay
13	320126510077	CHITTALA RADHA KRISHNA NAGA SAI	Radha Krishna
14	320126510078	D VENKATA SAI HEMANTH ASWIN	Aswin
15	320126510079	DASARI PRADEEP	Pradeep
16	320126510080	DODDI LIKHITHA	Likhitha
17	320126510081	DRAKSHARAPU LALITHA VENNELA	Vennela
18	320126510083	GOPISETTY AKHIL VENKAT	Akhil Venkat
19	320126510085	GUDLA SINDHUJA	Sindhuja
20	320126510086	GURRAM MOULI	Mouli
21	320126510087	ILLAPU ADARSH	Adarsh
22	320126510088	KAKARLA AASISH PRABHU	Asish
23	320126510089	KANDELLI RAJ KIRAN	Kiran
24	320126510090	KARRI VENKATA SATYA SAI RAJA REDDY	Raj Reddy
25	320126510091	KARROTHU HARI CHANDRA PRASAD	Prasad
26	320126510092	KHANDAVALLI MANOJ	Manoj
27	320126510093	KODA LAKSHMI SINDU BHARGAVI	Sindhu
28	320126510094	KOTANA DEEPIKA	Deepika
29	320126510095	LATCHIREDDI SAI CHARAN	Sai Charan
30	320126510096	MACHA HEMACHANDRA	Hemachandra
31	320126510097	MARADANA TANUJA	Tanuja
32	320126510098	MARAPAKULA MANASA	Manasa
33	320126510099	MARUMUDI HADASSA	Hadassa
34	320126510100	MOHAMMED ALTAF	Altaf
35	320126510101	MOHAN SAI NMANJUNADH	Nmanjunadh
36	320126510102	MYLIPILLI MAHESH	Mahesh
37	320126510103	NARAYANASETTI SAI LAVANYA	Sai Lavanya
38	320126510104	PALADUGU RUTHVIKH	Ruthvikh
39	320126510105	PALAKURTHI VYSHNAVI	Vyshnavi
40	320126510106	PALAVALASA VINAY KUMAR	Vinay Kumar
41	320126510107	PASUPELETI SONI	Soni
42	320126510108	PEDDAPATI RADHA KRISHNA SWAMY	Radha Krishna Swamy
43	320126510109	PEDIREDLA SAI GANESH	Sai Ganesh
44	320126510110	PENTAKOTA VIDYA MRUDULA	Vidya Mrudula
45	320126510111	PIDUGU MOUNIKA	Mounika

S.No	Roll No	Name of the student	Signature
46	320126510112	PRODDUTURI SOURABH BHARADWAJ	<i>Sourabh</i>
47	320126510113	PUSARLA PYDI VENKATA RAJESH	<i>Rajesh</i>
48	320126510114	PUVVALA SIRISHA	<i>Sirisha</i>
49	320126510115	REDDIPALLI GRUHA SATYA SAI VANAJA	<i>Sai Vanaja</i>
50	320126510116	ROKKAM NIKHILA	<i>Nikhila</i>
51	320126510117	SAPPA ALEKHYA	<i>Alekhy</i>
52	320126510118	SATTI UHENDRA VIJAY NAGA MANIKANTA REDDY	<i>Vijay</i>
53	320126510119	SENAPATHI PRASANNA	<i>Prasanna</i>
54	320126510120	SHAIK KHAIRUNNISA	<i>Khairunisa</i>
55	320126510121	SURISSETTY LAVANYA	<i>Lavanya</i>
56	320126510122	THOTA CHARAN KARTHIK	
57	320126510123	VANAPARTHI JYOTHIRMAI	<i>Jyothirmai</i>
58	320126510124	VAVILAPALLI VINOD	<i>Vinod</i>
59	320126510125	VEMULAKONDA VAMSI RAMAKRISHNA	<i>Vamsi</i>
60	320126510126	VENKATA SAI KAUSHIK ARAVELLI	<i>Kaushik</i>
61	320126510127	VERUPANDA VAMSI	<i>Vamsi</i>
62	320126510128	YASWANTH BEHARA	<i>Yaswanth</i>
63	321126510L07	SHAIK MUSKAN SULTANA	<i>Muskan</i>
64	321126510L08	NALLI KISHORE BABU	<i>Kishore</i>
65	321126510L09	YALAGADA PAVAN KUMAR	<i>Pavan</i>
66	321126510L10	IPPILI SANJITH	<i>Sanjith</i>
67	321126510L11	KANGANI GAYATHRI	<i>Gayatri</i>
68	321126510L12	MULAKALAPALLI ALEKYA	<i>Alekya</i>

*Dr. S. Suresh*

Head of the Department of  
Computer Science & Engineering  
Anil Neerukonda Institute of  
Technology & Sciences  
Sangivalasa, Visakhapatnam Dist

