

GE6161 - Computer Practices Lab

Manual

**(First semester B.E/B.Tech. Students for the
Academic Year 2015-16)**

www.studentsfocus.com

VALLIAMMAI ENGINEERING COLLEGE
SRM NAGAR, KATTANKULATHUR – 603 203

Prepared by

T. Rajasekaran, M.Tech.,
Asst. Professor (O.G)
Department of Computer Science & Engineering



GE6161 - Computer Practices Lab Manual

First Edition - August 2013

Second Edition - August 2014

Third Edition - August 2015

Instructions to the students

The following instructions must be followed by the students in their laboratory classes.

1. Students are expected to be punctual to the lab classes. If they are late, they will be considered absent for that particular session.
2. Students should strictly maintain the dress code.
3. Students must bring their observation note, record note (completed with previous experiment) to every lab class without fail.
4. Students are advised to come with full preparation for their lab sessions by
 - Reading the detailed procedure of the exercise from the laboratory manual.
 - Completion of observation note book (i.e.) Aim, Procedure, Algorithm, Program should be written and Flowchart should be drawn in the observation note before entering into the laboratory.
5. Students must use pen for writing and pencil to draw Flowchart in observation note book.
6. Students must get attestations immediately for their output/execution.
7. Students are advised to enter their results evaluated in the observation note book on the same day of that exercise.
8. Assessment marks for each exercise is based only on their performance in the laboratory.
9. Record note has to be completed then and there and get corrected when the students are coming for the next lab class.
10. Students must strictly maintain silence during lab classes.
11. If any of the students is absent for the lab class for genuine reasons, he/she will be permitted to do the exercise during the repetition class only.
12. If any student is found causing damage to the lab equipments, he/she shall replace the same with a new.



CONTENTS

S. No	EXERCISE	Page No
	Document Creation, Search, Generate, Manipulation data using MS Office	
1.	1.a) Preparation of an Advertisement in MS-Word	1
	1.b) Preparation of Curriculum Vitae (CV) in MS-Word	5
	1.c) Creation of Scientific Notations in MS-Word	9
	1.d) Mail Merge and Letter preparation in MS-Word	11
	1.e) Presentation about a Company using MS-PowerPoint	15
	Presentation and Visualization – Graphs, Charts, 2D,3D	
2.	2.1) Graphs and Charts in 2D, 3D using MS-Excel	17
	2.2) Student Marks List analysis using formula in MS-Excel	21
	2.3) Calculation of Salary details in MS-Excel	27
	Problem formulation, Problem solving and Flowcharts	
3.	3.1) Drawing Flowcharts – Biggest among three numbers	29
	3.2) Drawing Flowcharts – Fibonacci series	31
	3.3) Drawing Flowcharts – Sum of digits of an integer	33
	C Programming using Simple statements and expressions	
4.	4.1) Finding the Roots of a Quadratic Equation	35
	4.2) Armstrong number	39
	4.3) Menu driven program	43
	Scientific problem using decision making and looping	
5.	5.1) Generating Prime Numbers between a given ranges	49
	5.2) Generating Sine series	53
	5.3) Reversing the digits of a number	57
	Simple programming for one dimensional and two dimensional arrays	
6.	6.1) Ascending and Descending order of the given numbers	61
	6.2) Matrix Multiplication	65
	6.3) String Palindrome	71
	Solving problems using String functions	
7.	7.1) String manipulation	75
	7.2) Sorting strings	79
	7.3) String operations	83
	Programs with user defined functions – includes parameter passing	
8.	8.1) Swapping of two numbers - call by value	87
	8.2) Swapping of two numbers - call by reference	91
	Program using recursive functions	
9.	9.1) Combinatorics (${}_n C_r$) using recursion	95
	9.2) Fibonacci series up to a given term using recursion	99
	Program using structures and unions	
10.	10.1) Students records using structure	103
	10.2) Payroll processing using union	109
	Annexure I - Additional C Programs for exercise	113
	Annexure II - C Questions and Answers	133



MICROSOFT OFFICE EXERCISES

Document Creation, Search and Text Manipulation using MS -Word

EX.NO : 1.A PREPARATION OF AN ADVERTISEMENT IN MS-WORD

DATE :

AIM:

To prepare an advertisement for a company with the following specifications using MS-WORD.

- a. Attractive Page border
- b. Use at least one Clip art
- c. Design the name of the company using Word art
- d. Give details of the Company in brief
- e. Use bullets if necessary

PROCEDURE:

Step 1: Open a new word document using **File** → **New** option.

Step 2: Go to **Page Layout** → **Page Borders**, under the Page border tab choose the appropriate style and color, etc, and click OK.

Step 3: Go to **Page Layout** → **Page Color**, choose the appropriate color.

Step 4: Go to **Insert** → **Clip Art**, search for a relevant picture from the collections and insert it in to the page.

Step 5: Go to **Insert** → **Word Art**, choose the appropriate style from the list, type the company name and click OK.

Step 6: Type the company details in the document and do the following steps for various styles.

- a. For **bullets and numbering**, select the appropriate style from the menu and apply to the paragraph.



- b. For **alignment**, select the particular word or statement or paragraph in the document then press **CTRL+L** (left) or **CTRL+R** (right) or **CTRL+E** (center) or **CTRL+J** (justify) or select the relevant button from the formatting toolbar.



- c. For **bold facing**, select the particular word or statement or paragraph then press **CTRL+B** or select the on the formatting toolbar.
- d. For **italic style**, select the particular word then press **CTRL+I** or select the on the formatting toolbar.

OUTPUT:

VALLIAMMAI ENGINEERING COLLEGE

SRM Nagar, Kattankulathur – 603203

VALLIAMMAI ENGINEERING COLLEGE, a Division of **SRM Group of Educational Institutions** for privileged Learning, is functioning under the auspices of Valliammai Educational Society (which provides quality education in the various fields for more than 3 decades) with the noble vision of, **“Educate to excel in social Transformation”** since 09.09.1999.

The College is approved by **AICTE, New Delhi** and affiliated to **Anna University, Chennai, Tamil Nadu**.

The college offering 3 programmes namely **Engineering & Technology, Management and Computer Applications**. In this 7 Under Graduate courses and 8 Post Graduate courses are offered by VEC.

UG Courses	PG Courses
❖ B.E – Civil Engineering	❖ M.E – Structural Engineering
❖ B.E – Computer Science & Engineering	❖ M.E – Communication Systems
❖ B.E – Electronics & Communication Engg.	❖ M.E – Computer Science & Engineering
❖ B.E – Electrical & Electronics Engineering	❖ M.E – Power Systems Engineering
❖ B.E – Electronics & Instrumentation Engg.	❖ M.E – Control & Instrumentation Engg
❖ B.E – Mechanical Engineering	❖ M.E – CAD/CAM
❖ B.Tech – Information Technology	❖ M.C.A – Master of Computer Applications
	❖ M.B.A – Master of Business Administration

For Admissions

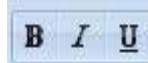


Valliammai Engineering College
SRM Nagar, Kattankulathur - 603203,
Kancheepuram Dt, Tamil Nadu.

TEL : 044 - 27454784 / 726, FAX : 044 - 27451504.

Email : srmvec@valliammai.co.in

- e. For **underlining**, select the particular word or statement or paragraph then press **CTRL+U** or select the on the formatting toolbar.



- f. For **spelling corrections**, select the particular word, then right click the mouse or click spelling and grammar button on the standard tool bar.



- g. For **saving the document**, press **CTRL+S** or click on save button on standard tool bar or select the save option from the File menu.



The above formatting options are generally used for creating the document. Apart from these, MS-WORD provides a bunch of formatting options for making the document as an eye-catching one.



RESULT:

Thus an advertisement for a company with the relevant specifications was prepared using MS-WORD and the output was verified.

OUTPUT:**A. Rahul****CURRICULUM VITAE****CONTACTS:**

No. 5, 1st Floor,
3rd Avenue,
Anna Nagar,
Chennai – 600060,
Tamil Nadu.

91- 98765 43210
arahul@valliammai.co.in

OBJECTIVE:

To attain a challenging position in the IT field with my skills and abilities and to achieve professional growth.

EDUCATIONAL QUALIFICATIONS:

Course	College/ School	Year	Aggregate
B.E (Computer Science and Engg)	Valliammai Engineering College, Anna University, Chennai	2012	9.6 CGPA
HSC	Chennai Matriculation School, Chennai.	2008	95 %
SSLC	Chennai Matriculation School, Chennai.	2006	90 %

SKILL SET:

Operating Systems : MS Windows, Redhat Linux
Languages : C, C++, JAVA
Databases : Ms Access, MySQL

AREAS OF INTEREST:

- Data Structures
- Networks
- Data Mining

EX.NO: 1.B PREPARATION OF CURRICULUM VITAE (CV) IN MS-WORD**DATE :****AIM:**

To create a two pages of curriculum vitae (CV) of a graduate with the following specifications using MS-WORD.

- a. Table to show qualifications with proper headings
- b. Appropriate left and right margins
- c. Page number in the footer on the right side
- d. Name on each page on the top right side

PROCEDURE:

Step 1: Open a new word document using **File** → **New** option.

Step 2: Go to **Page Layout** → **Margins**, under the Margins tab choose the appropriate document margin from the list or customize using **Custom Margins** and click OK.

Step 3: Go to **Insert** → **Header or Footer**, choose the relevant style and type the necessary details to display in every page of the document.

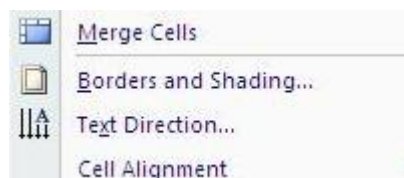
Step 4: Go to **Insert** → **Page Number**, choose the relevant number format to display in every page of the document.



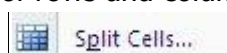
Step 5: Go to **Insert** → **Table**, and specify the required number of rows and columns.

Step 6: Type the required information, and do the following for various operations.

- a. To change the text direction, select the cell, right click, select the **Text Direction** and choose the text orientation.
- b. To change the text alignment, select the cell, right click, select the **Cell Alignment** and choose the relevant alignment from the list.
- c. To combine cells, select the cells that are to be merged then right click and select **Merge Cells** from the menu.



- d. To split a cell, select the cell, right click and select **Split Cells** from the menu and give the no. of rows and columns as required.



A. Rahul

ACADEMIC PROJECT:

Project title: Online Calligraphic Manuscript Recognition

Description: Automatic identification of handwritten script facilitates many important applications such as automatic transcription of multilingual documents and search for documents on the web containing a particular script. The increase in the usage of handheld devices which accept handwritten input has created a growing demand for algorithms that can efficiently analyze and retrieve handwritten data. Online documents may be written in different languages and scripts. A single document page in itself may contain text written in multiple scripts. The main aim of this project is Recognition of six different manuscripts (i.e., Arabic, Cyrillic, Devnagari, Han, Hebrew and Roman).

CO-CURRICULAR ACTIVITIES:

- Presented Seminar on "Cloud computing", Wireless LAN Security"
- Participated various spot events Debugging, Quiz etc., in various colleges.

CERTIFICATE OF HONOUR:

- Participated in Athletes in the Divisional Level and secured Certificate.
- Secured "Best Player" award in college.

PERSONAL PROFILE:

Date of Birth, Age : 28th April 1991, 21
Gender : Male
Marital Status : Single
Languages known : English & Tamil.
Father's name : S. Arun
Hobbies : Listening to Music, Planting.

Declaration:

I A. RAHUL hereby declare that the details I have furnished above are true to the best of my knowledge.

Authentically,

Place: Chennai

Date:

(A. ARUL)

- e. For **bullets and numbering**, select the appropriate style from the menu and apply to the paragraph.



- f. To apply borders & shading, select the cell, rows, or columns then right click, choose the **Borders and Shading** from the menu and add the relevant borders, shading and color.

Step 7: Save the document by pressing **CTRL+S** or click on save button on standard tool bar or select the save option from the File menu.



RESULT:

Thus curriculum vitae (CV) was created with the relevant specifications using MS-WORD and the output was verified.

L

OUTPUT:



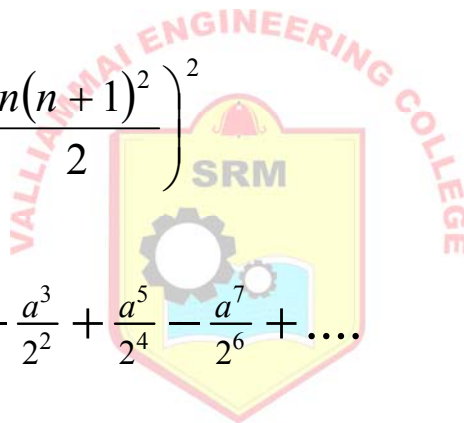
$$(a + b)^3 = a^3 + b^3 + 3a^2b + 3b^2a$$

$$X = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a}$$

$$\sum_{i=1}^n i^2 = \frac{n(n+1)(2n+1)}{6}$$

$$\sum_{i=1}^n i^3 = \left(\frac{n(n+1)}{2} \right)^2$$

$$X = a - \frac{a^3}{2^2} + \frac{a^5}{2^4} - \frac{a^7}{2^6} + \dots$$



EX.NO : 1.C CREATION OF SCIENTIFIC NOTATIONS IN MS-WORD

DATE :

AIM:

To create the following scientific notations using MS-WORD.

a. $Ca(HCO_3)_2 = CaCO_3 + CO_2 + H_2O$

b. $(a+b)^3 = a^3 + b^3 + 3a^2b + 3b^2a$

c. $X = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a}$

d. $\sum_{i=1}^n i^2 = \frac{n(n+1)(2n+1)}{6}$

e. $\sum_{i=1}^n i^3 = \left(\frac{n(n+1)}{2} \right)^2$

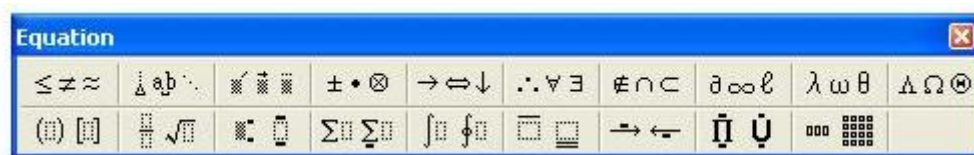
f. $X = a - \frac{a^3}{2^2} + \frac{a^5}{2^4} - \frac{a^7}{2^6} + \dots$

PROCEDURE:

Step 1: Open a new word document using **File** → **New** option.

Step 2: To create Equation, choose **Insert** → **Object** → **Microsoft Equation 3.0** and click OK.

Step 3: Type the formulae from the equation toolbar according to the given specification.



Step 4: **Save** the document by pressing **CTRL+S** or click on save button on standard tool bar or select the save option from the File menu.

RESULT:

Thus the scientific notations wer created by using MS-WORD and the output was verified.

L

OUTPUT:

No. VEC/Principal/2013/Col.Opening/177

18.07.2013

To

Mr./Ms. R.Rahul
10A, Anna Nagar,
Chennai - 60,

Dear Student,

Sub: I Year B.E/B.Tech. Course – Opening of College – Intimation - reg.

I happy to note that you have secured admission in our college. I welcome you to attend the regular classes which will commence from **01.08.2013 (Thursday)**. You are informed to report to this college office on 01.08.2013 at 9.00 a.m.

PRINCIPAL



No. VEC/Principal/2013/Col.Opening/177

18.07.2013

To

Mr./Ms. A.Ramya
No.5, Stanly Road,
Trichy - 621301,

Dear Student,

Sub: I Year B.E/B.Tech. Course – Opening of College – Intimation - reg.

I happy to note that you have secured admission in our college. I welcome you to attend the regular classes which will commence from **01.08.2013 (Thursday)**. You are informed to report to this college office on 01.08.2013 at 9.00 a.m.

PRINCIPAL

EX.NO: 1.D MAIL MERGE AND LETTER PREPARATION IN MS-WORD**DATE :****AIM:**

To prepare a letter template to inform the students about the reopening date of the college by using Mail merge in MS-WORD.

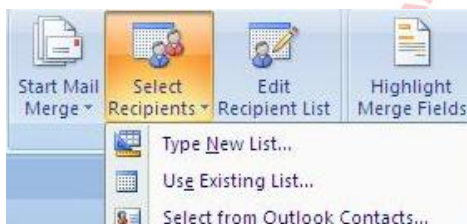
PROCEDURE:

Step 1: Open a new word document using **File** → **New** option.

Step 2: To start mail merge, go to **Mailings** → **Start Mail Merge** and select type of the merge you wish to create from the list. For example, Letter.



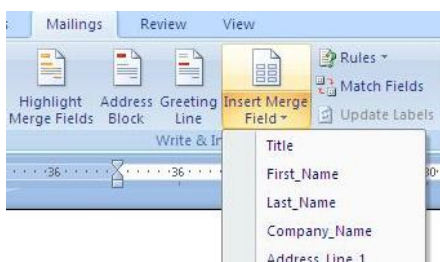
Step 3: To create recipients, select the **Select Recipients** → **Type New List**.



A new window will appear, type the recipient's information and click on the button OK and then save the data source. (select **Use Existing List** to use the existing data source).



Step4: To insert field in to the document, click the button **Insert Merge Field** and select the field you wish to insert. (Repeat the step 4 until you have inserted all required fields.)



L

OUTPUT:

No. VEC/Principal/2013/Col.Opening/177

18.07.2013

To

Mr./Ms. B.Chandran

25, Tambaram,

Chennai - 45,

Dear Student,

Sub: I Year B.E/B.Tech. Course – Opening of College – Intimation - reg.

I happy to note that you have secured admission in our college. I welcome you to attend the regular classes which will commence from **01.08.2013 (Thursday)**. You are informed to report to this college office on 01.08.2013 at 9.00 a.m.

PRINCIPAL



No. VEC/Principal/2013/Col.Opening/177

18.07.2013

To

Mr./Ms. G.Kumar

22 B Block, Coimbatore Main Road,

Coimbatore - 73,

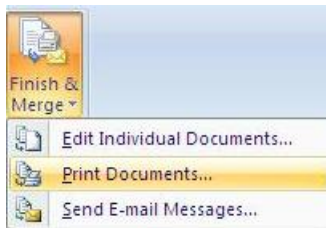
Dear Student,

Sub: I Year B.E/B.Tech. Course – Opening of College – Intimation - reg.

I happy to note that you have secured admission in our college. I welcome you to attend the regular classes which will commence from **01.08.2013 (Thursday)**. You are informed to report to this college office on 01.08.2013 at 9.00 a.m.

PRINCIPAL

Step5: Finally, to create or print the letter, click on the button **Finish & Merge**, from the list appear select **Print Documents**.



The **Merge or Printer** window will appear then select the radio button **All** and click on the button OK. The **Printer** window will appear select the printer you wish to use and click OK. You can also send this letter to all the recipients by email by selecting **Finish & Merge** → **Send E-mail Message**.

Step 6: Save the document by pressing **CTRL+S** or click on save button on standard tool bar or select the save option from the File menu.



RESULT:

Thus the letter template was prepared using Mail merge in MS-WORD and the output was verified.











OUTPUT:



Prince India Overview

- Among the leading recruitment companies in India
- Been in business successfully for over 25 years
- Offices in 8 cities across India
- Over 1,00,000 candidates placed successfully
- ISO 9001:2000 certified for Quality Management Systems
- Registered with the Ministry of Labor & Saudi Consulate
- Associated with DBS Centers across India
- Preferred recruiter with India's largest job portals Naukri.com & Timesjobs.com

Our Expertise

 CONSTRUCTION & ENGINEERING	 HOSPITALITY
 FACILITIES MANAGEMENT	 INFORMATION TECHNOLOGY
 HEAVY INDUSTRIES & MANUFACTURING	 HEALTHCARE
 OIL & GAS	 RETAIL

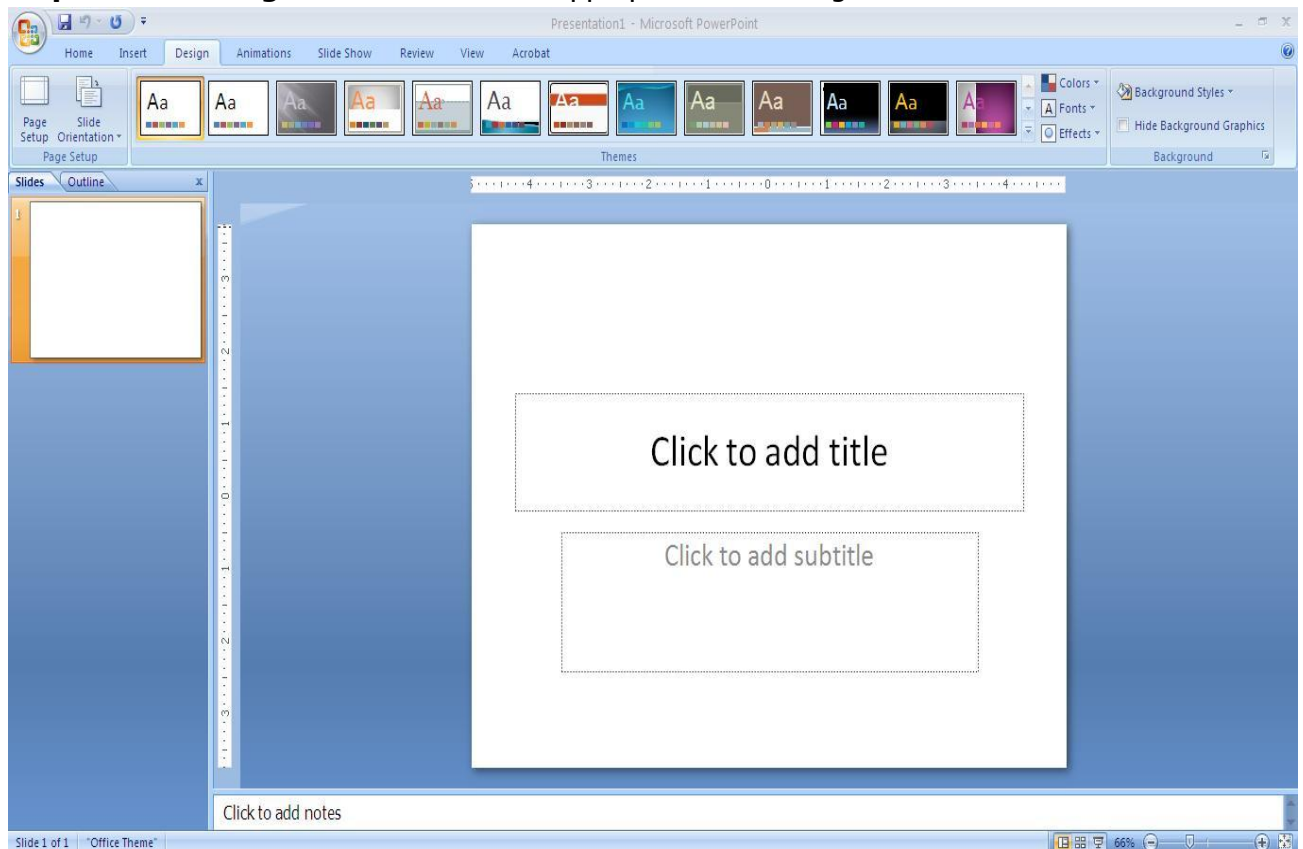
EX.NO : 1.E PRESENTATION ABOUT A COMPANY IN MS-POWERPOINT**DATE :****AIM:**

To create presentation about a company using MS-PowerPoint.

PROCEDURE:

Step 1: Open a new PowerPoint document (Start→Programs→Microsoft Office→Microsoft PowerPoint 2007)

Step 2: Go to **Design tab** and select the appropriate slide design from the list.



Step 3: Click where it says "Click to add title" and type the title of your presentation.

Step 4: Click where it says "Click to add subtitle" and type a secondary title or topic subtitle.

Step 5: To add a new slide click on Home→New Slide.

Step 6: Go to **Insert** → **Clip Art**, search for a relevant picture from the collections and insert it in to the slide.

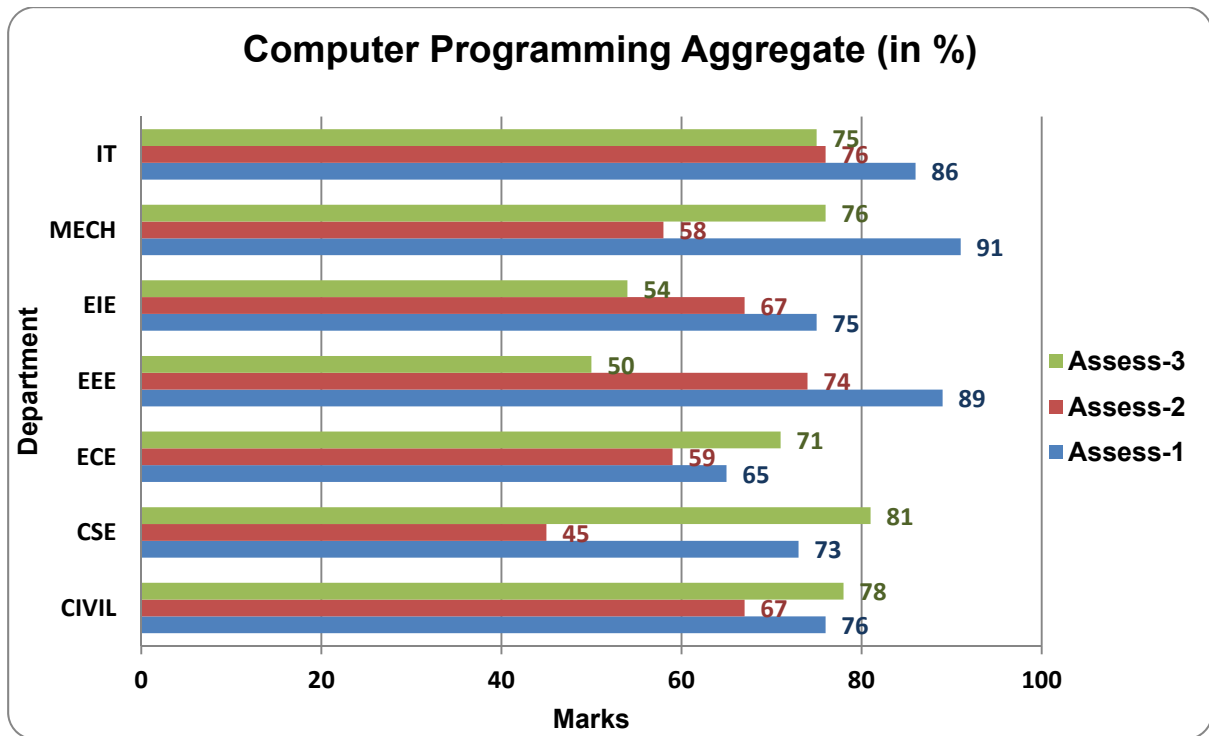
Step 7: Go to **Insert** → **Picture**, choose the relevant picture from the

RESULT:

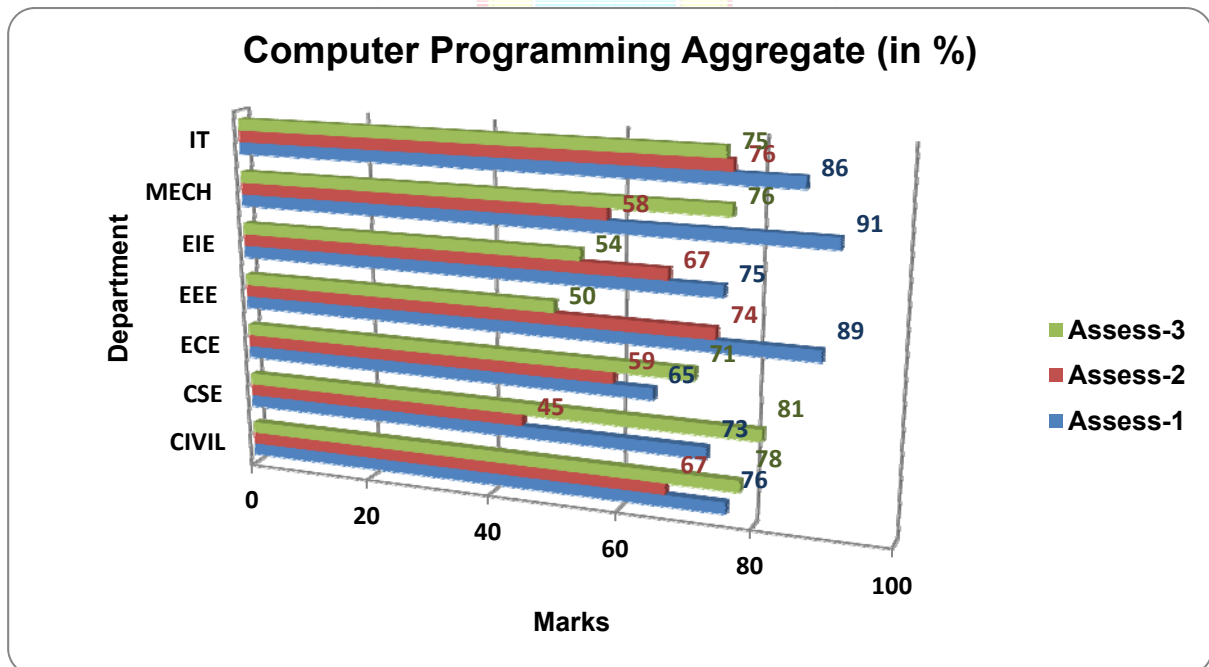
Thus the presentation about a company was prepared in MS-POWERPOINT and the output was verified.



OUTPUT:



2D Bar Chart



3D Bar Chart

MICROSOFT OFFICE EXERCISES**Presentations and Visualizations – Graphs, Charts using MS-Excel****EX.NO: 2.1 GRAPHS AND CHARTS IN 2D, 3D USING MS-EXCEL****DATE :****AIM:**

To create a spreadsheet and implement the graphs and charts in 2D & 3D using MS-Excel.

- a. Create bar chart for the following :

Computer Programming Aggregate (in %)			
Department	Assess-1	Assess-2	Assess-3
CIVIL	76	67	78
CSE	73	45	81
ECE	65	59	71
EEE	89	74	50
EIE	75	67	54
MECH	91	58	76
IT	86	76	75

- b. Create line chart for the following :

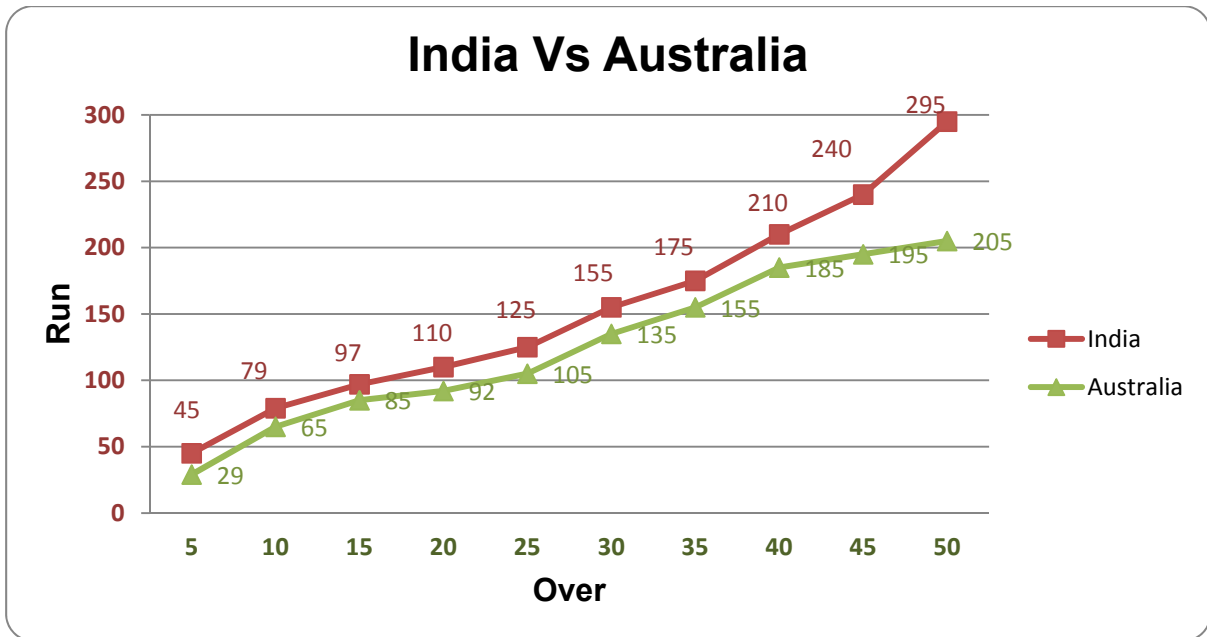
India Vs Australia		
Overs	India	Australia
5	45	29
10	79	65
15	97	85
20	110	105
25	125	132
30	155	150
35	175	180
40	210	215
45	240	238
50	295	281

- c. Create Pie chart for the following :

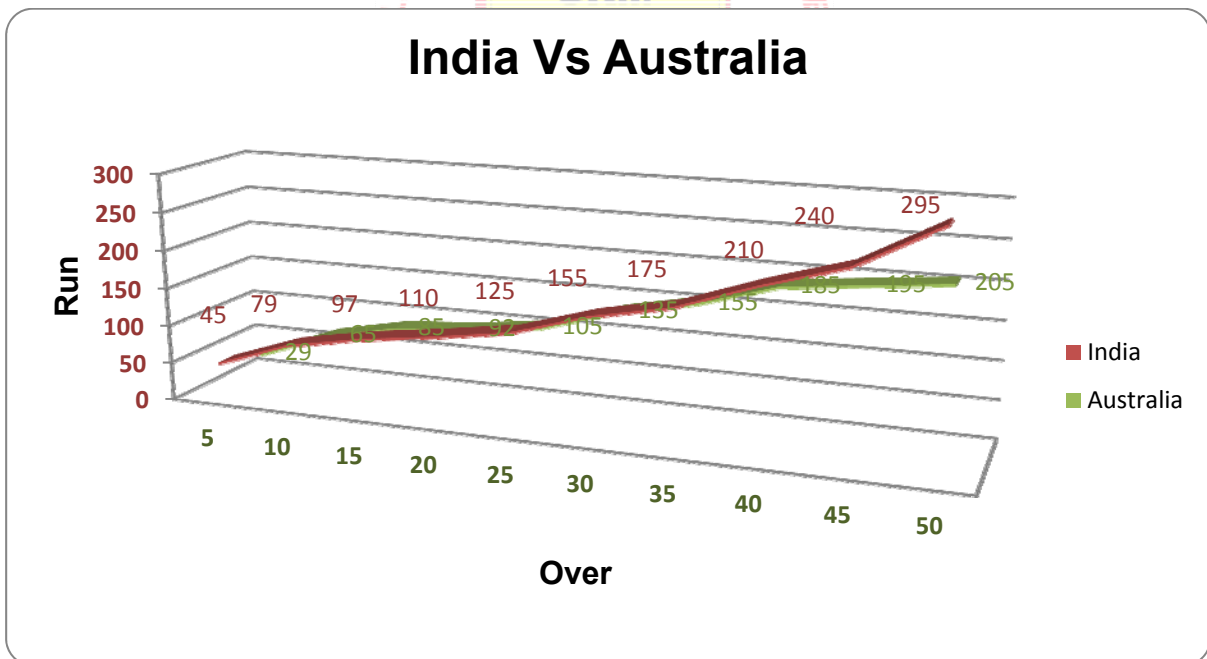
First Quarter Sales	
Product	Sales(in Crores)
Printers	234
Scanner	90
Fax	109
Hubs	150
Desktops	300
Laptops	270
Servers	100
Accessories	450



OUTPUT:



2D Line Chart



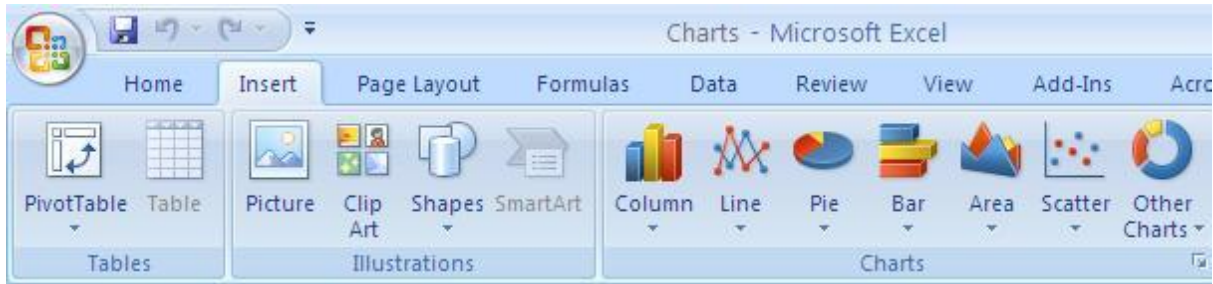
3D Line Chart

PROCEDURE:

Step 1: Open a new spreadsheet document using **File** → **New** option.

Step 2: Enter the details in the Spread sheet for which the chart to be created.

Step 3: To create chart, select the data go to **Insert** → **Charts** and choose the appropriate chart type (Bar or Line or Pie, etc.,) from the list.



Step 4: To change the chart layout, go to **Design** → **Chart Layouts** then choose the appropriate chart layout style and enter the chart title, horizontal and vertical axis.



Step 5: To add data labels in the chart, go to **Layout** → **Data Labels** and select the relevant position.

Step 6: To change the Axis data range, select the chart then go to **Design** → **Select Data**, and edit in the **Select Data Source Dialog** box.

Step 7: **Save** the document by pressing **CTRL+S** or click on save button on standard tool bar or select the save option from the File menu.

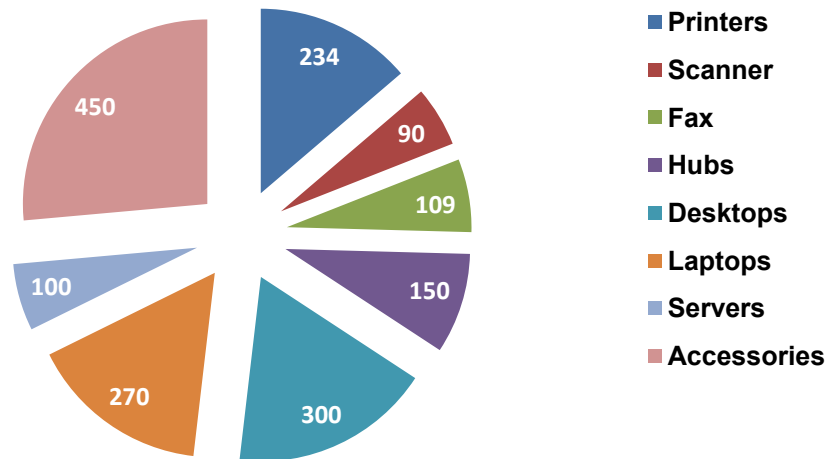
RESULT:

Thus the spreadsheet was created to implement the graphs and charts in 2D and 3D using MS-Excel and the output was verified.



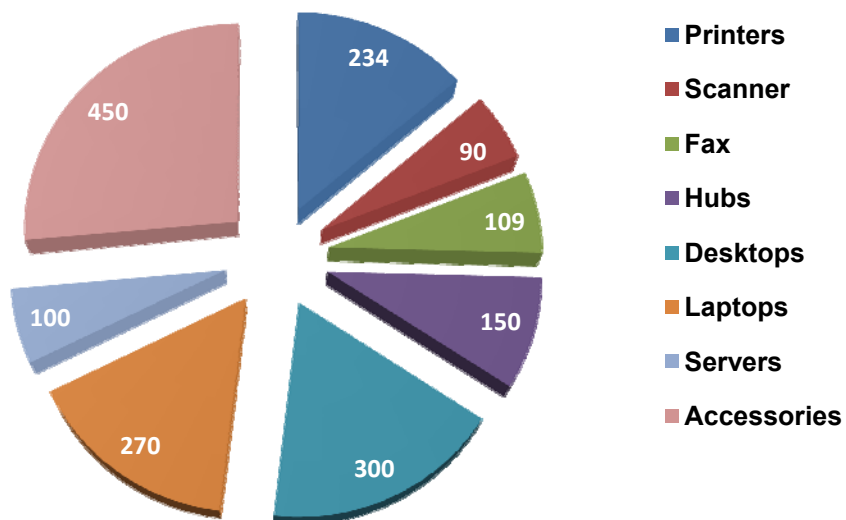
OUTPUT:

First Quarter SalesSales(in Crores)



2D Pie Chart

First Quarter SalesSales(in Crores)



3D Pie Chart

EX.NO : 2.2 STUDENT MARKS LIST ANALYSIS USING FORMULA IN MS-EXCEL**DATE :****AIM:**

To determine result, rank, grade and cut-off of students in the student mark lists in the following format and also sort Rank in ascending order using MS-Excel.

S.No	Name	Lang	Eng	Maths	Phy	Che	CS	Total	Avg	Result	Rank	Grade	Cut off
1.	Raj	80	90	100	100	95	80						
2.	Arun	80	95	65	95	70	75						
3.	Prem	65	85	95	85	65	95						
4.	Kumar	45	75	80	85	65	50						
5.	Siva	85	65	70	65	85	85						
6.	Prabhu	95	85	55	45	65	75						
7.	Anu	65	80	85	65	75	90						
8.	Bala	75	65	95	85	80	75						
9.	Jagan	55	45	65	75	65	55						
10.	Raji	75	65	75	70	80	65						

* The Rank, Grade and Cut-off marks of a student must be calculated if he/she scored a minimum mark (i.e., mark ≥ 50) in all the subjects.

* Finally, sort the students list based on the Rank.

Legend:

Result: PASS or FAIL

Rank : Descending order (1, 2, 3...)

Grade :

- ❖ If obtained average mark is more than or equal to 90, and less than 100, then grade would be "S".
- ❖ If obtained average mark is more than or equal to 80, and less than 90, then grade would be "A".
- ❖ If obtained average mark is more than or equal to 70, and less than 80, then grade would be "B".
- ❖ If obtained average mark is more than or equal to 60, and less than 70, then grade would be "B".
- ❖ If obtained average mark is less than 60, then grade would be "F " - Failed.

Cut off: out of 100. (Maths/2 + Phy/4 + Che/4).

L

OUTPUT:

Row /Col	A	B	C	D	E	F	G	H	I	J	K	L	M	N
1	S.No	Name	Lang	Eng	Maths	Phy	Che	CS	Total	Avg	Result	Rank	Grade	Cut off
2	1	Raj	80	90	100	100	95	80	545	90.83	PASS	1	S	98.75
3	2	Arun	80	95	65	95	70	75	480	80	PASS	3	A	73.75
4	3	Prem	65	85	95	85	65	95	490	81.67	PASS	2	A	85.00
5	4	Kumar	45	75	80	85	65	50	400	66.67	FAIL		F	
6	5	Siva	85	65	70	65	85	85	455	75.83	PASS	6	B	72.50
7	6	Prabhu	95	85	55	45	65	75	420	70	FAIL		F	
8	7	Anu	65	80	85	65	75	90	460	76.67	PASS	5	B	77.50
9	8	Bala	75	65	95	85	80	75	475	79.17	PASS	4	B	88.75
10	9	Jagan	55	45	65	75	65	55	360	60	FAIL		F	
11	10	Raji	75	65	75	70	80	65	430	71.67	PASS	7	B	75.00

Final Result (before sorting descending order - Rank)

PROCEDURE:

Step 1: Open a new spreadsheet document using **File** → **New** option.

Step 2: Enter all the details in the worksheet.

Step 3: To calculate the **Total** of the first student, use the formula =**SUM(C2:H2)**. Then drag that formula for entire records.

SUM(Cell number1, Cell number2,) or

SUM(Cell number1: Cell number N) in the series (Range).

Step 4: To calculate the **Average**, use the formula =**AVERAGE(C2:H2)**. Then drag that formula for entire records.

AVERAGE(Cell number1, Cell number2,) or

AVERAGE(Cell number1: Cell number N) in the series (Range).

Step 5: To calculate the **Result**, use the following formula. Then drag that formula for entire records.

=IF(COUNTIF(C2:H2,"<50"),"FAIL","PASS").

- *COUNTIF(range, criteria)* returns a value that satisfies the criteria in the range.
- *IF(logical test, [value if true], [value if false])* returns the true value if the condition is satisfied or returns the false value.

Step 6: To calculate the **Rank** of the first student's total in cell I2, enter this formula in cell L2. Then, copy the formula from cell L2 down to cell L11, and the total will be ranked in descending order.

=IF(K2="FAIL","",RANK(I2,\$I\$2:\$I\$11,0))

The above formula first checks whether the Result is PASS or FAIL. If the result is FAIL, then it returns a null value, else calculates the Rank by using the formula RANK(I2,\$I\$2:\$I\$11,0).

The syntax for the RANK function is:

RANK(number, array, [order])

number is the number to find the rank for.

array is a range or array of numbers to use for ranking purposes.

order is optional. It specifies how to rank the numbers.

If order is 0, it ranks numbers in *descending order*.

If order is 1, it ranks numbers in *ascending order*.

(We want to compare the number to the list of numbers in cells \$I\$2:\$I\$11. Use an **absolute reference** (\$I\$2:\$I\$11), instead of a **relative reference** (I2:I11) so the referenced range will stay the same when you copy the formula down to the cells below.)

L

OUTPUT:

Row /Col	A	B	C	D	E	F	G	H	I	J	K	L	M	N
1	S.No	Name	Lang	Eng	Maths	Phy	Che	CS	Total	Avg	Result	Rank	Grade	Cut off
2	1	Raj	80	90	100	100	95	80	545	90.83	PASS	1	S	98.75
3	3	Prem	65	85	95	85	65	95	490	81.67	PASS	2	A	85.00
4	2	Arun	80	95	65	95	70	75	480	80	PASS	3	A	73.75
5	8	Bala	75	65	95	85	80	75	475	79.17	PASS	4	B	88.75
6	7	Anu	65	80	85	65	75	90	460	76.67	PASS	5	B	77.50
7	5	Siva	85	65	70	65	85	85	455	75.83	PASS	6	B	72.50
8	10	Raji	75	65	75	70	80	65	430	71.67	PASS	7	B	75.00
9	4	Kumar	45	75	80	85	65	50	400	66.67	FAIL		F	
10	6	Prabhu	95	85	55	45	65	75	420	70	FAIL		F	
11	9	Jagan	55	45	65	75	65	55	360	60	FAIL		F	

Final Result (after sorting descending order - Rank)

Step 7: To calculate the **Grade** of the first student's average in cell J2, enter this formula in cell M2. Then, copy the formula from cell M2 down to cell M11, and the grade will be shown as S, A, B and F.

=IF(K2="FAIL","F",IF(J2>=90,"S",IF(J2>=80,"A",IF(J2>=70,"B","C"))))

The above formula first checks whether the Result is PASS or FAIL. If the result is FAIL, then it returns a null value, else calculates the Grade by using the formula =IF(J2>=90,"S",IF(J2>=80,"A",IF(J2>=70,"B","C"))) and returns a value as follows.

- ❖ Returns "S" grade if Average mark is ≥ 90 & < 100 .
- ❖ Returns "A" grade if Average mark is ≥ 80 & < 90 .
- ❖ Returns "B" grade if Average mark is ≥ 70 & < 80 .
- ❖ Returns "C" grade if Average mark is ≥ 60 & < 70 .
- ❖ Returns "F" grade if Average mark is < 60 .

Step 8: To calculate the **Cut-off mark** of the first student's marks in cell E2, F2 and G2, enter this formula in cell N2. Then, copy the formula from cell N2 down to cell N11, and the cut-off mark will be calculated for all the students in the list

=IF(K2="FAIL","",E2/2+F2/4+G2/4)

The above formula first checks whether the Result is PASS or FAIL. If the result is FAIL, then it returns a null value, else calculates the Cut-off marks by using the formula =E2/2+F2/4+G2/4.

Step 8: Finally, after all calculations are done, select the Rank [column L] in the worksheet, then go to **Data** → **Sort**, choose the type of sorting order from the Sort dialog box and click OK.

Step 9: Save the document by pressing **CTRL+S** or click on save button on standard tool bar or select the save option from the File menu.

RESULT:

Thus the spreadsheet was created to manipulate the details according to the given conditions and formats using MS Excel and the output was verified.

L

OUTPUT:

Row/Col	A	B	C	D	E	F	G	H	I	J	K
1	S.No	Name	Basic Pay	HRA	DA	TA	Gross Salary	PF	LIC	Ded.	Net Salary
2	1	Raj	20,000	3,600	3,000	2,400	29,000	2,000	1,400	3,400	25,600
3	2	Arun	15,000	2,700	2,250	1,800	21,750	1,500	1,050	2,550	19,200
4	3	Prem	12,000	2,160	1,800	1,440	17,400	1,200	840	2,040	15,360
5	4	Kumar	10,000	1,800	1,500	1,200	14,500	1,000	700	1,700	12,800
6	5	Siva	13,000	2,340	1,950	1,560	18,850	1,300	910	2,210	16,640
7	6	Prabhu	11,000	1,980	1,650	1,320	15,950	1,100	770	1,870	14,080
8	7	Anu	18,000	3,240	2,700	2,160	26,100	1,800	1,260	3,060	23,040
9	8	Bala	19,000	3,420	2,850	2,280	27,550	1,900	1,330	3,230	24,320
10	9	Jagan	14,000	2,520	2,100	1,680	20,300	1,400	980	2,380	17,920
11	10	Raji	17,000	3,060	2,550	2,040	24,650	1,700	1,190	2,890	21,760

Result (before sorting in alphabetical order)

Row/Col	A	B	C	D	E	F	G	H	I	J	K
1	S.No	Name	Basic Pay	HRA	DA	TA	Gross Salary	PF	LIC	Ded.	Net Salary
2	7	Anu	18,000	3,240	2,700	2,160	26,100	1,800	1,260	3,060	23,040
3	2	Arun	15,000	2,700	2,250	1,800	21,750	1,500	1,050	2,550	19,200
4	8	Bala	19,000	3,420	2,850	2,280	27,550	1,900	1,330	3,230	24,320
5	9	Jagan	14,000	2,520	2,100	1,680	20,300	1,400	980	2,380	17,920
6	4	Kumar	10,000	1,800	1,500	1,200	14,500	1,000	700	1,700	12,800
7	6	Prabhu	11,000	1,980	1,650	1,320	15,950	1,100	770	1,870	14,080
8	3	Prem	12,000	2,160	1,800	1,440	17,400	1,200	840	2,040	15,360
9	1	Raj	20,000	3,600	3,000	2,400	29,000	2,000	1,400	3,400	25,600
10	10	Raji	17,000	3,060	2,550	2,040	24,650	1,700	1,190	2,890	21,760
11	5	Siva	13,000	2,340	1,950	1,560	18,850	1,300	910	2,210	16,640

Result (after sorting in alphabetical order)

EX.NO : 2.3 CALCULATION OF SALARY DETAILS IN MS-EXCEL**DATE :****AIM:**

To calculate HRA, DA, TA, PF, LIC, Gross salary, Deduction, Net salary from the given data in the worksheet and also sort all name in alphabetical order using MS-Excel.

HRA= 18% of basic pay

TA= 12% of Basic pay

DA=15% of basic pay

PF= 10% of basic pay

LIC=7% of basic pay

Gross Salary= basic pay + HRA + DA + TA

Deduction = PF + LIC

Net salary = Gross Salary – Deduction

PROCEDURE:

Step 1: Open a new spreadsheet document using **File** → **New** option.

Step 2: Enter the employee names and their basic pay in the worksheet.

Step 3: To calculate the **HRA** of an employee, use the formula **=C2*0.18**. Then drag that formula for entire records.

Step 4: To calculate the **DA** of an employee, use the formula **=C2*0.15**. Then drag that formula for entire records.

Step 5: To calculate the **TA** of an employee, use the formula **=C2*0.12**. Then drag that formula for entire records.

Step 6: To calculate the **Gross Salary** of an employee, use the formula **=SUM(C2:F2)**. Then drag that formula for entire records.

SUM(Cell number1, Cell number2,) or
SUM(Cell number1: Cell number N) in the series (Range).

Step 7: To calculate the **PF** of an employee, use the formula **=C2*0.1**. Then drag that formula for entire records.

Step 8: To calculate the **LIC** of an employee, use the formula **=C2*0.07**. Then drag that formula for entire records.

Step 9: To calculate the **Deduction** of an employee, use the formula **=SUM(H2:I2)**. Then drag that formula for entire records.

Step 10: To calculate the **Net Salary** of an employee, use the formula **=G2-J2**. Then drag that formula for entire records.

Step 8: Finally, after all calculations are done, select the Name [column B] in the worksheet, then go to **Data** → **Sort**, choose ascending sorting order from the Sort dialog box and click OK.

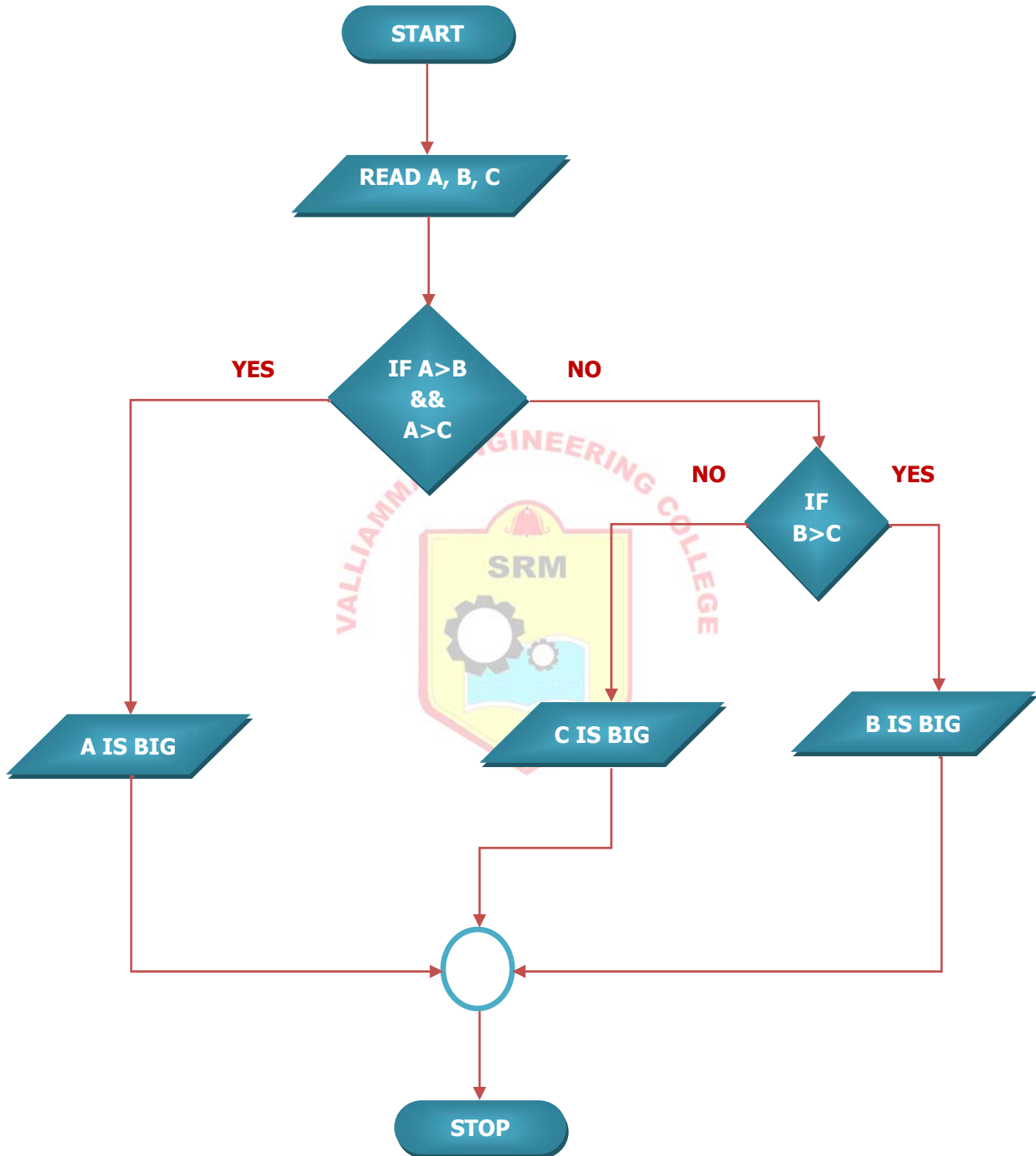
Step 9: **Save** the document by pressing **CTRL+S** or click on save button on standard tool bar or select the save option from the File menu.

RESULT: Thus the salary details were computed from the given set of data and also sorted all name in alphabetical order and the output was verified.

L

OUTPUT:

Flowchar to find biggest among three numbers



MICROSOFT OFFICE EXERCISES**Problem Formulation, Problem Solving and Flowcharts****EX.NO : 3. DRAWING FLOWCHARTS****DATE :****AIM:**

To draw a flowcharts for the following using MS-WORD.

1. To find the biggest among three integers.
2. To generate a Fibonacci series up to a specified number.
3. To find the sum of the digits of an integer.

PROCEDURE:

Step 1: Open a new word document using **File** → **New** option.

Step 2: To select a flowchart symbol, select **Insert** → **Shapes** → **Flowchart** option.

Step 3: To insert a symbol in to the document, click anywhere in the document and left click the mouse. Repeat step 2 & 3 for each symbols that you want to add.

Step 4: To **add text** in to the symbol, select the symbol then do right click the mouse and select Add text and type the text.

Step 5: To connect different symbols, select Arrows, Lines from **Insert** → **Shapes**.

Step 6: To **Group/ Ungroup** the flowchart symbols and lines in the document, select the symbols that you want to group by **Home** → **Select Objects** then do Right click the mouse and select **Grouping** → **Group/Ungroup** from the menu.

Step 7: **Save** the document by pressing **CTRL+S** or click on save button on standard tool bar or select the save option from the File menu.

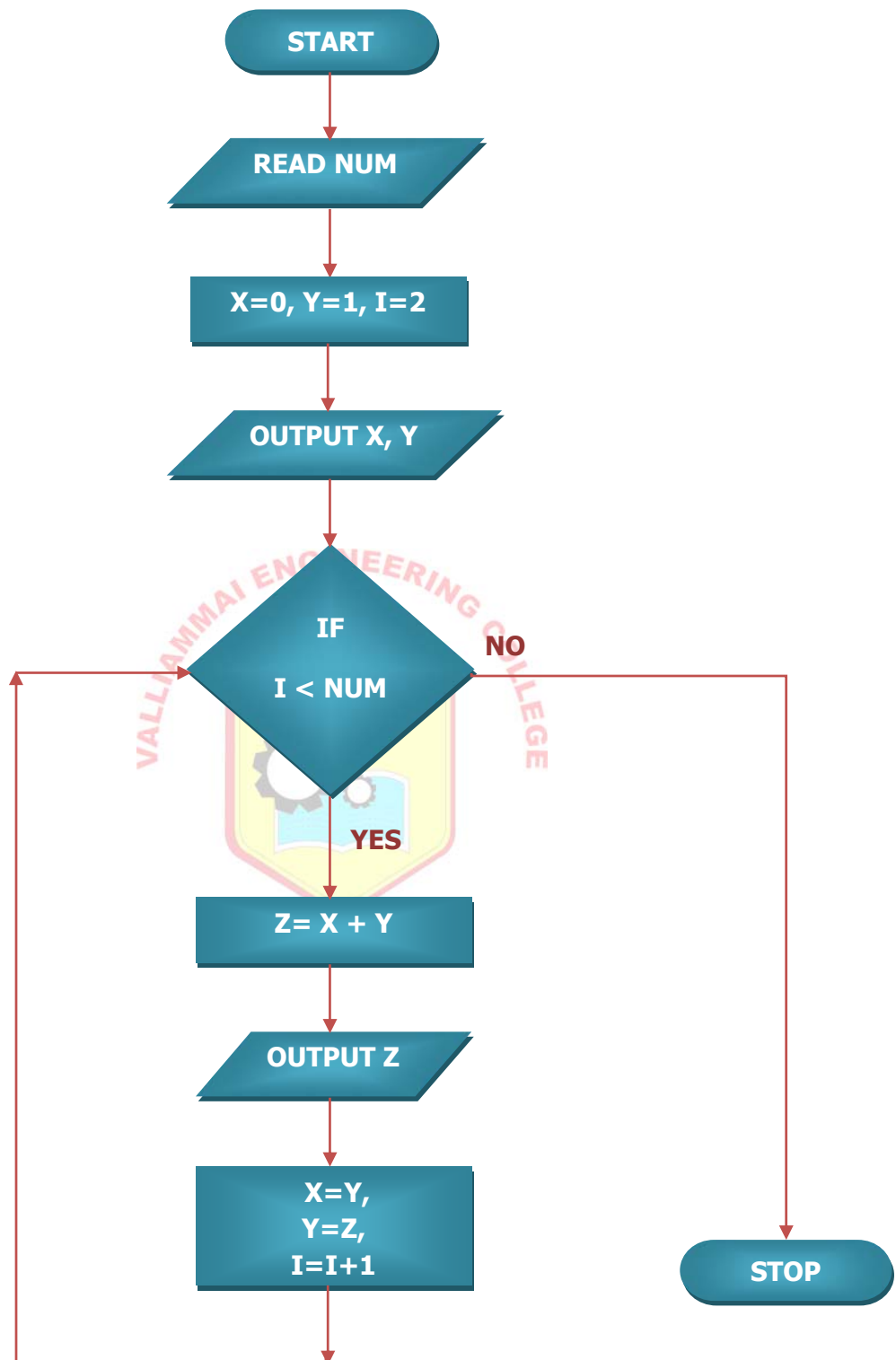
3.1 Algorithm: To find the biggest among three integers

- | | |
|-----------|---------------------------|
| Step-1: | Start |
| Step-2: | Read the variable A, B, C |
| Step-3: | Check IF(A >B && A>C) |
| Step-3.1: | Print A is BIG. |
| Step-4: | Otherwise, check IF(B>C) |
| Step-4.1: | Print B is BIG |
| Step-5: | else Print C is BIG |
| Step-6: | Stop. |



OUTPUT:

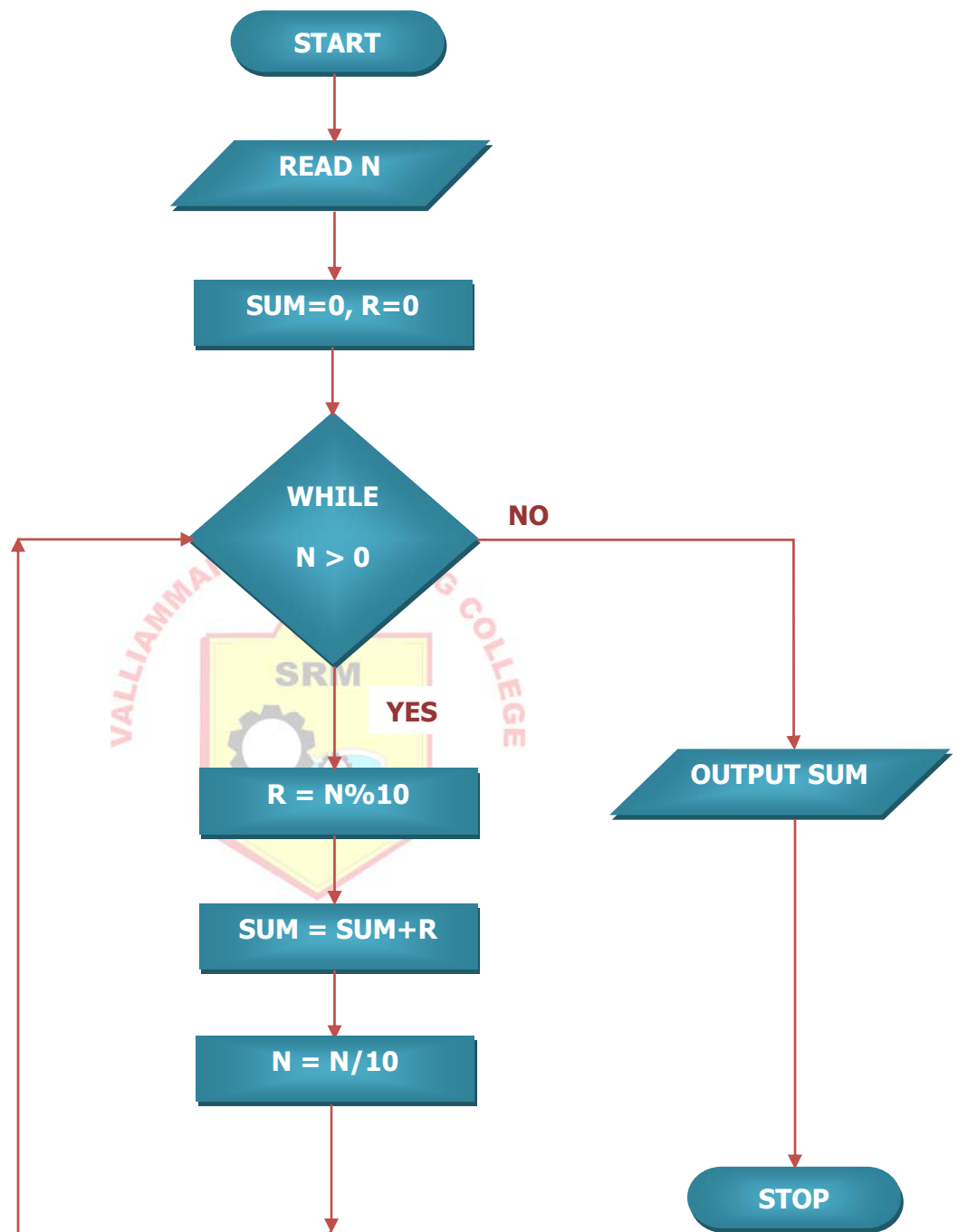
Flowchart to generate a Fibonacci series up to a specified number



L

OUTPUT:

Flowchart to find the sum of the digits of an integer



3.3 Algorithm: To find the sum of the digits of an integer

- Step-1: Start
- Step-2: Read the variable N
- Step-3: Assign the variable SUM=0, R=0
- Step-3: Set a loop using the condition WHILE(N>0), If Yes
 - R = N%10;
 - SUM = SUM+R;
 - N = N/10;
- Step-4: Print Z;
- Step-5: Else, Exit
- Step-6: Stop.

Explanation:

For example, if N =123,

PROCESS	Loop 1	Loop 2	Loop 3	Loop 4
GET N SUM=0, R=0 WHILE (N>0) { R = N%10; SUM=SUM+R; N= N/10; } PRINT SUM EXIT;	NUM = 123 SUM=0, R=0 WHILE (123>0) { R=123%10=3; SUM=0+3=3; N=123/10=12; } PRINT SUM EXIT;	WHILE (12>0) { R=12%10=2; SUM=3+2=5; N=12/10=1; } PRINT SUM EXIT;	WHILE (1>0) { R=1%10=1; SUM=5+1=6; N=1/10=0; }	WHILE (0>0) { <i>Condition not satisfied</i> } ELSE EXIT;
Result	SUM = 3	SUM = 5	SUM =6	Nil

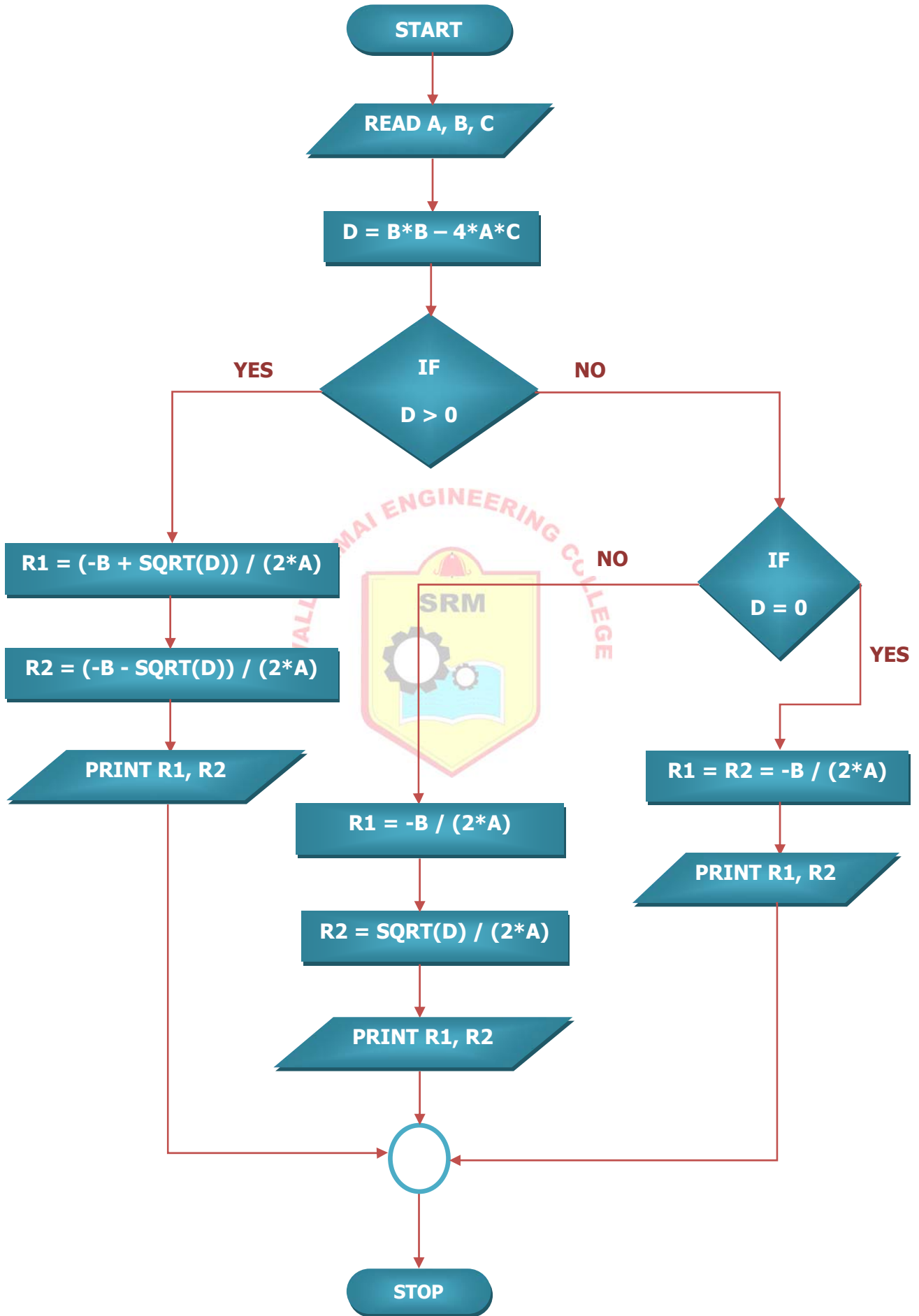
Final output: 6

RESULT:

Thus the flowcharts were generated using MS-WORD and the output was verified.

L

FLOWCHART:



C PROGRAMMING EXERCISES**C Programming using Simple statements and expressions****EX.NO : 4.1 FINDING THE ROOTS OF QUADRATIC EQUATION****DATE :****AIM:**

To write a C Program to find the roots of a Quadratic equation.

ALGORITHM:

- Step 1: Start
- Step 2: Read the variable a, b, c.
- Step 3: Compute $d = b^2 - 4ac$.
- Step 4: Test the condition, d is greater than 0 using IF statement.
Step 4.1: Calculate: $r1 = \frac{-b + \sqrt{d}}{2a}$.
Step 4.2: Calculate: $r2 = \frac{-b - \sqrt{d}}{2a}$.
Step 4.3: Print the roots r1 and r2.
- Step 5: Else, test the condition, d is equal to 0 using IF statement.
Step 5.1: Calculate: $r1=r2 = -b / (2a)$.
Step 5.2: Print the roots r1 and r2.
- Step 6: Else, compute real and imaginary as
Step 6.1: Calculate: $real = -b / (2a)$.
Step 6.2: Calculate $imag = \sqrt{-d}/(2a)$.
Step 6.3: Print the real and imag.
- Step 7: Stop

L

OUTPUT:

```
D:\WINDOWS\system32\cmd.exe - tc
PRG TO FIND THE ROOTS OF A QUADRATIC EQUATION
-----
Enter the coefficients a, b and c:
2.3 4 5.6
Roots are: -0.87+1.30i and -0.87-1.30i. They are complex.

PRG TO FIND THE ROOTS OF A QUADRATIC EQUATION
-----
Enter the coefficients a, b and c:
4 1 0
Roots are: 0.00 and -0.25. They are real and distinct.
```



PROGRAM:

```

#include <stdio.h>
#include <math.h> /* This is needed to use sqrt() function.*/
#include <conio.h>
void main()
{
float a, b, c, d, r1,r2, real, imag;
clrscr();
printf("\nPRG TO FIND THE ROOTS OF A QUADRATIC EQUATION");
printf("\n-----");
printf("Enter the coefficients a, b and c: ");
scanf("%f%f%f",&a,&b,&c);
d=b*b-4*a*c; /*calculating the value of d */
if (d>0) /*Checking real solution is possible or not */
{
r1= (-b+sqrt(d))/(2*a); /*finding the root of quadratic equation */
r2= (-b-sqrt(d))/(2*a);
printf("Roots are: %.2f and %.2f. They are real and distinct.", r1 , r2);
}
else if (d==0)
{
r1 = r2 = -b/(2*a);
printf("Roots are: %.2f and %.2f. They are real and equal.", r1, r2);
}
else
{
real= -b/(2*a); /*calculating real and imaginary */
imag = sqrt(-d)/(2*a);
printf("Roots are: %.2f+%.2fi and %.2f-%.2fi. They are complex.", real, imag,
real, imag);
}
getch();
}

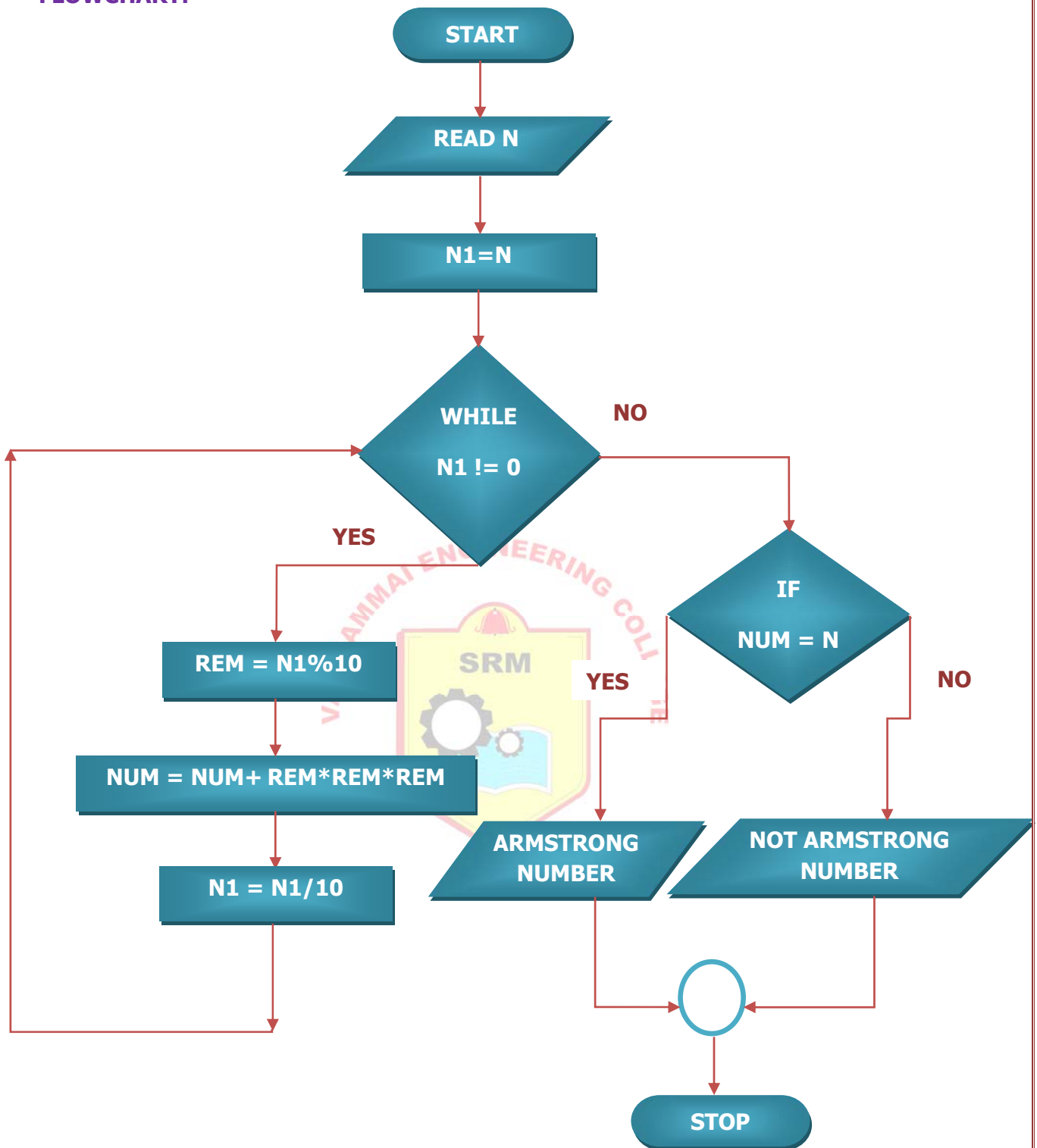
```

RESULT:

Thus the C program for finding roots of quadratic equation was executed and output was obtained.

L

FLOWCHART:

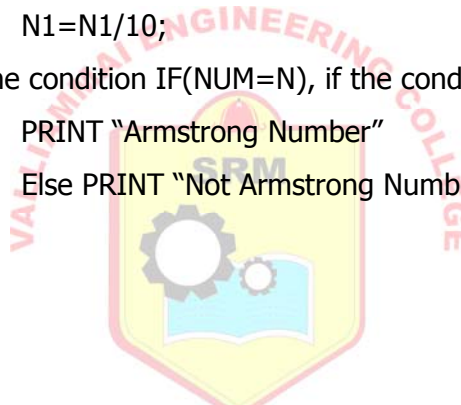


EX.NO : 4.2**ARMSTRONG NUMBER****DATE :****AIM:**

To write a C Program to check whether the given number is Armstrong or not.

ALGORITHM:

- Step 1: Start
- Step 2: Read the variable N
- Step 3: Assign $N1=N$;
- Step 4: Set a loop using the condition $WHILE(N1!=0)$, if the condition true
- Step 4.1: $REM=N1\%10$;
- Step 4.2: $NUM=NUM+REM*REM*REM$;
- Step 4.3: $N1=N1/10$;
- Step 5: Else, check the condition $IF(NUM=N)$, if the condition true
- Step 5.1 PRINT "Armstrong Number"
- Step 5.2 Else PRINT "Not Armstrong Number"
- Step 6: Stop



L

OUTPUT:

```
D:\WINDOWS\system32\cmd.exe - tc
PRG TO CHECK WHETHER A GIVEN NO. IS ARMSTRONG NO. OR NOT
-----
Enter a positive integer: 153
153 is an Armstrong number.

PRG TO CHECK WHETHER A GIVEN NO. IS ARMSTRONG NO. OR NOT
-----
Enter a positive integer: 371
371 is an Armstrong number.

PRG TO CHECK WHETHER A GIVEN NO. IS ARMSTRONG NO. OR NOT
-----
Enter a positive integer: 101
101 is not an Armstrong number.
```



PROGRAM:

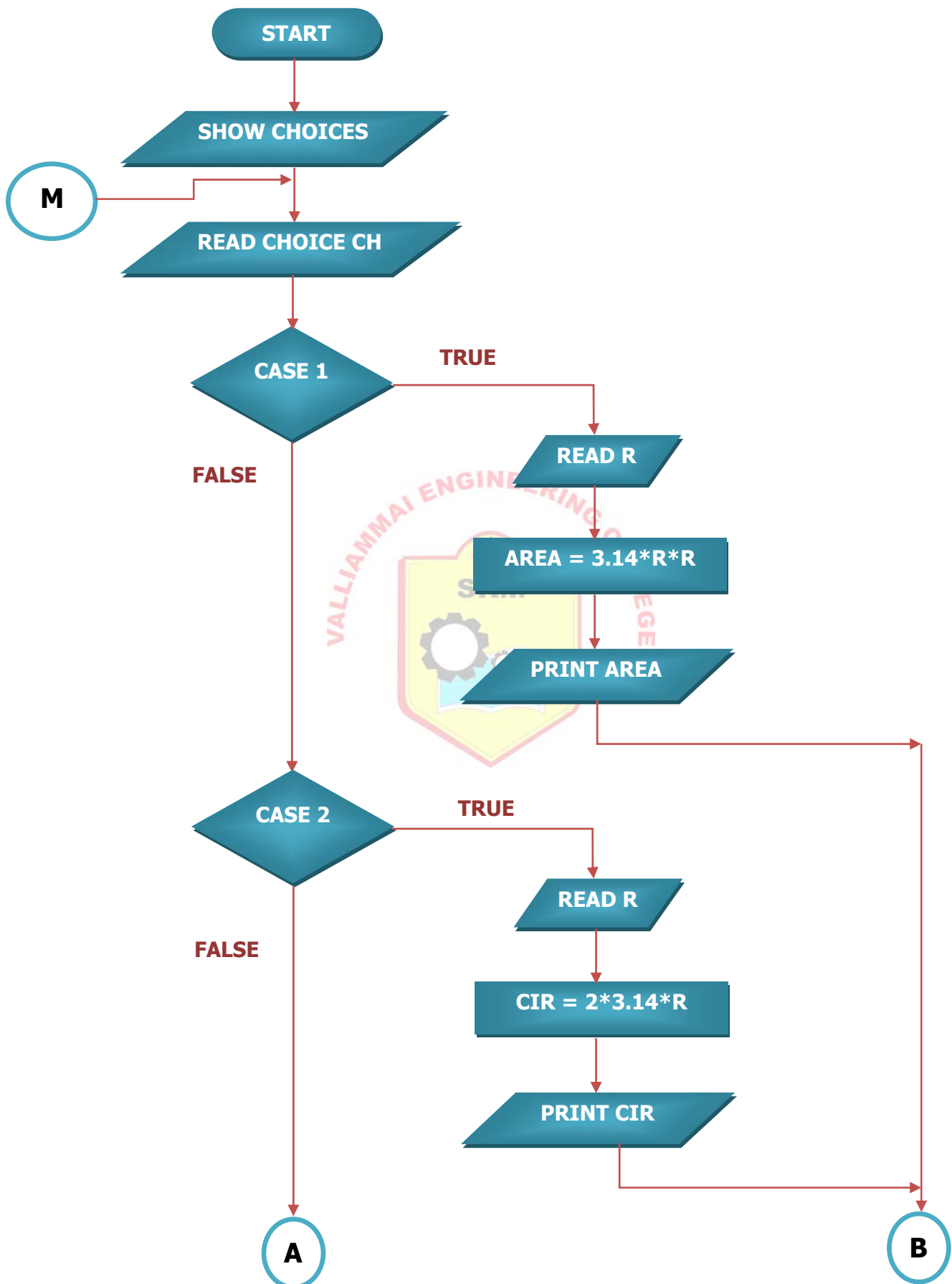
```
#include <stdio.h>
#include <conio.h>
void main()
{
    int n, n1, rem, num=0;
    clrscr();
    printf("\nPRG TO CHECK WHETHER A GIVEN NO. IS ARMSTRONG NO. OR NOT");
    printf("\n-----");
    printf("Enter a positive integer: ");
    scanf("%d", &n);
    n1=n;
    while(n1!=0)
    {
        rem=n1%10;
        num=num+(rem*rem*rem);
        n1=n1/10;
    }
    if(num==n)
        printf("%d is an Armstrong number.",n);
    else
        printf("%d is not an Armstrong number.",n);
    getch();
}
```

RESULT:

Thus the C program to check whether the given number is Armstrong or not was executed and the output was obtained.

L

FLOWCHART:



EX.NO: 4.3**MENU DRIVEN PROGRAM****DATE :****AIM:**

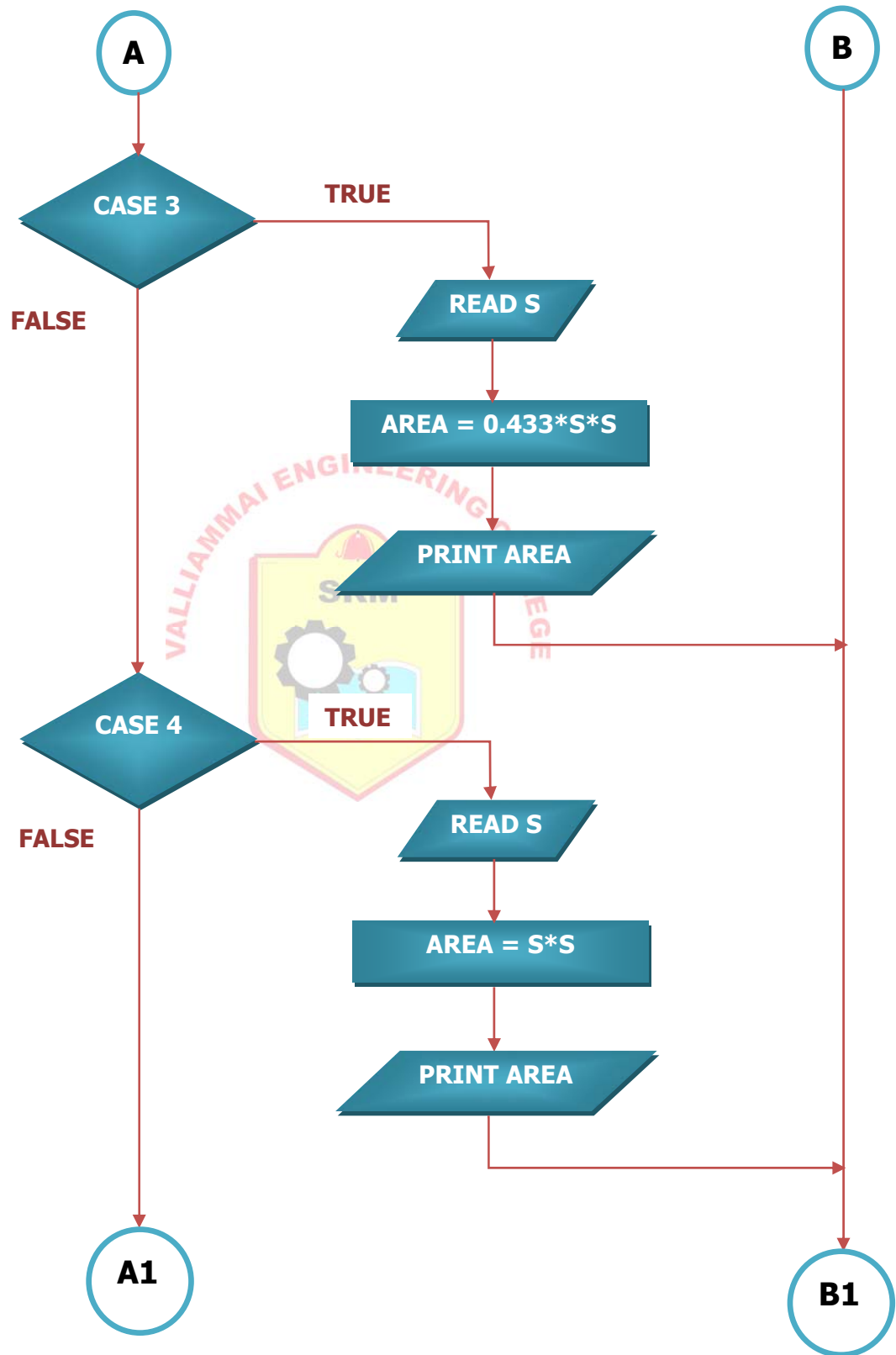
To write a simple menu driven C program using switch statement to find

- a. Area of a Circle
- b. Circumference of a Circle
- c. Area of a Equilateral triangle
- d. Area of a square
- e. To find odd or even,

ALGORITHM:

- Step 1: Start
- Step 2: Declare the variables
- Step 3: Show the choices and get the choice from the user
- Step 4: Pass the choice in to switch case
- Step 5: In case 1,
- Step 5.1: Get an input from the user
- Step 5.2: Calculate AREA = $3.14*r*r$
- Step 5.3: Print AREA
- Step 6: In case 2,
- Step 6.1: Get an input from the user
- Step 6.2: Calculate CIRCUMFERENCE = $2* 3.14*r$
- Step 6.3: Print CIRCUMFERENCE
- Step 7: In case 3,
- Step 7.1: Get an input from the user
- Step 7.2: Calculate AREA = $0.433*s*s$
- Step 7.3: Print AREA
- Step 8: In case 4,
- Step 8.1: Get an input from the user
- Step 8.2: Calculate AREA = $s*s$
- Step 8.3: Print AREA
- Step 9: In case 5,
- Step 8.1: Get a number from the user
- Step 8.2: Check IF($NUM\%2=0$)
- Step 8.3: If Yes, Print EVEN NUMBER
- Step 8.4: Else, PRINT ODD NUMBER
- Step 10: Check the Choice. If CH!=6 then go to step 3 else got to step 11.
- Step 11: Stop

L

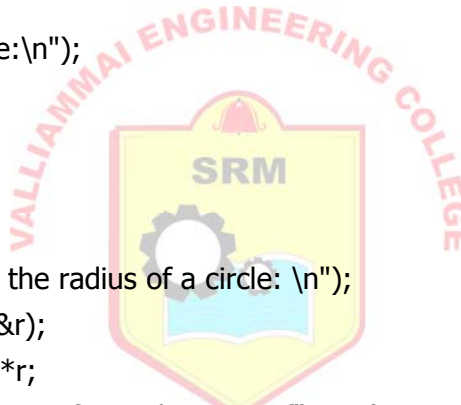


PROGRAM:

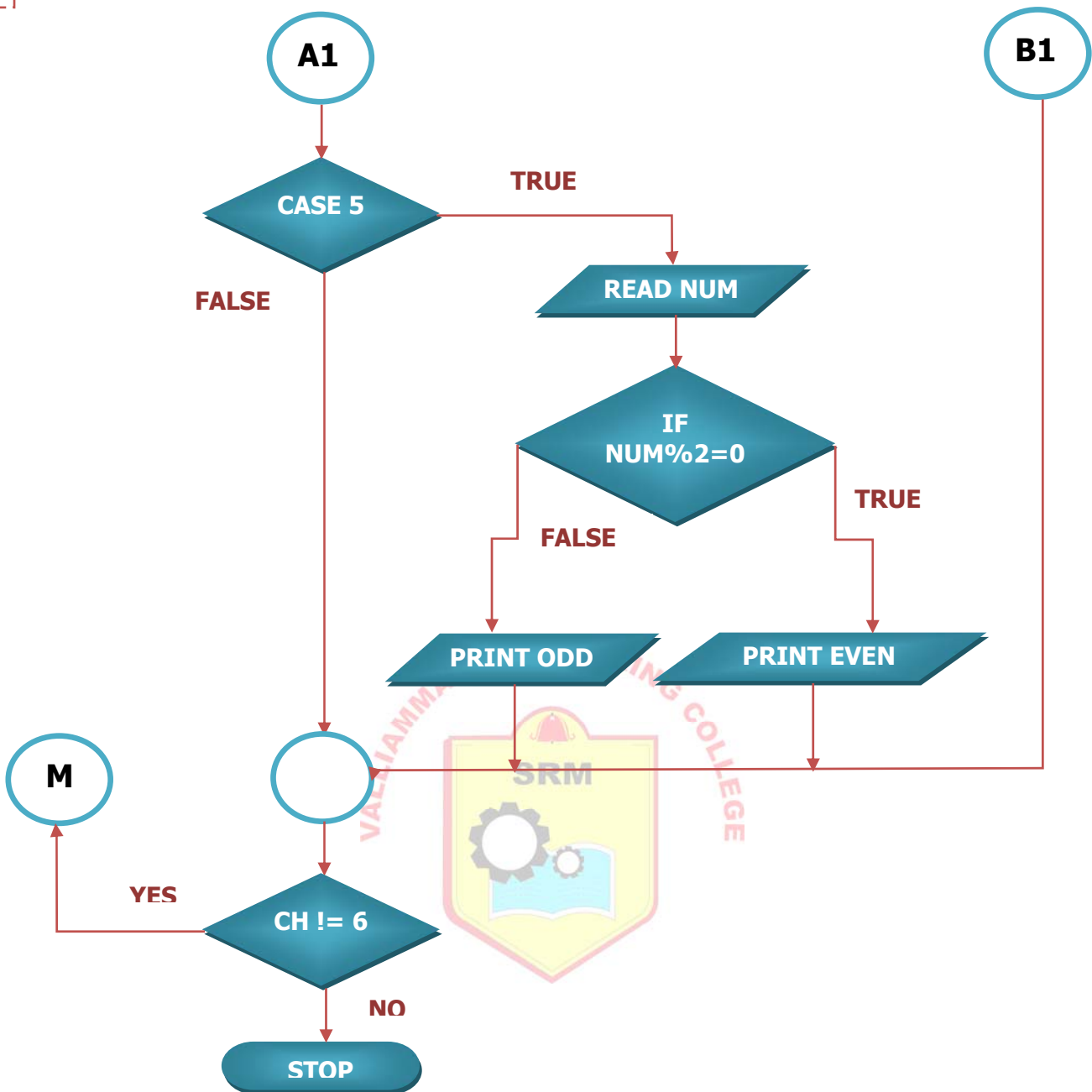
```
#include<stdio.h>
void main ()
{
    int ch, r, a, s, num;
    float area, cir;
    clrscr();
    do
    {
        printf("\n Choose any one of the options given below");
        printf("\n1. Area of a circle.\n");
        printf("2. Circumference of a circle.\n");
        printf("3. Area of an equilateral triangle.\n");
        printf("4. Area of a square.\n");
        printf("5. To find odd or even number.\n");
        printf("6. Exit.\n");
        printf("Enter your choice:\n");
        scanf("%d",&ch);
        switch(ch)
        {
            case 1:
                printf("Enter the radius of a circle: \n");
                scanf("%d",&r);
                area=3.14*r*r;
                printf("The area of a circle is=%.2f",area);
                break;

            case 2:
                printf("Enter the radius of a circle: \n");
                scanf("%d",&r);
                cir=2*3.14*r;
                printf("The circumference of circle is=%f",cir);
                break;

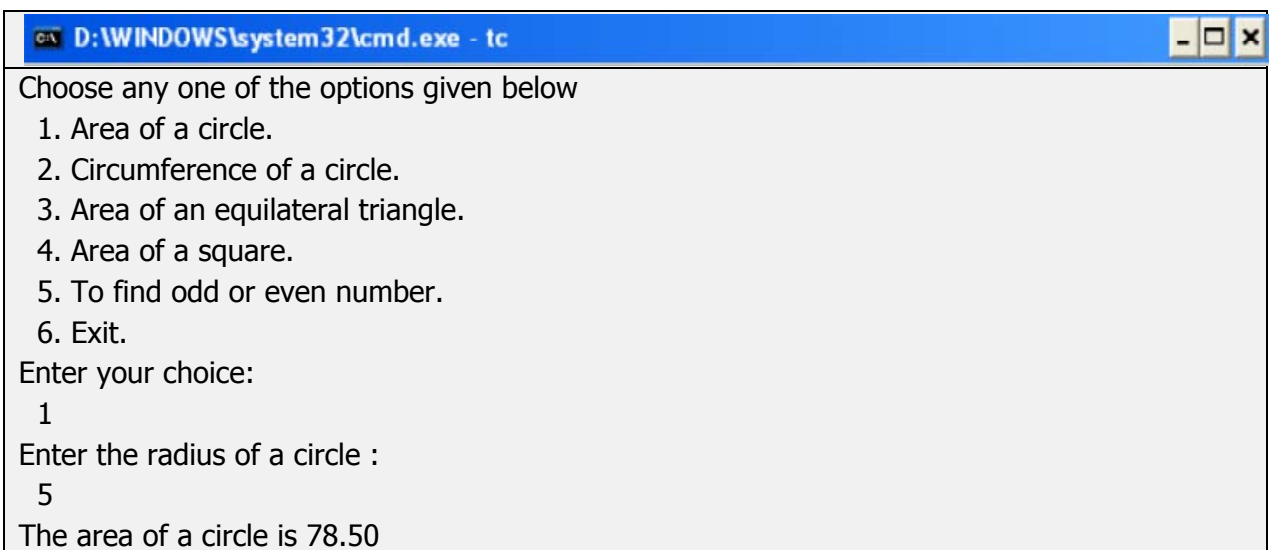
            case 3:
                printf("Enter the length of a side of an equilateral triangle: \n");
                scanf("%d",&s);
                area= 0.433*s*s;
                printf("The area of an equilateral triangle is=%f",area);
                break;
```



L



OUTPUT:



case 4:

```
printf("Enter the length of a side of a square: \n");
scanf("%d",&s);
area=s*s;
printf("The area of a square is=%f",area);
break;
```

case 5:

```
printf("Enter a number: \n");
scanf("%d",&num);
if(num%2==0)
    printf("\n%d is an even number",num);
else
    printf("\n%d is an odd number",num);
break;
```

case 6: exit();

}

}

while (ch!=6);

getch();

}

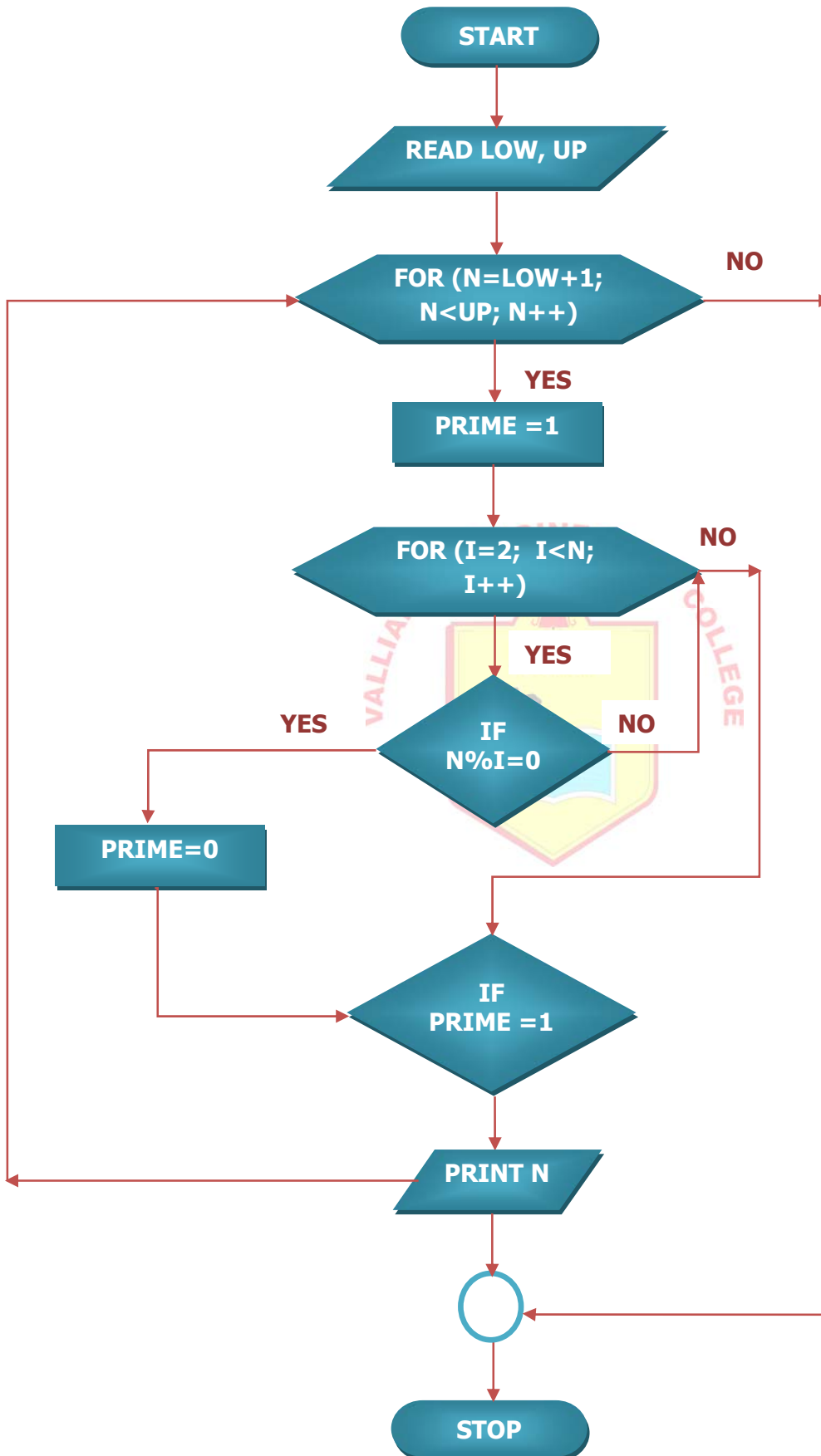


RESULT:

Thus a simple menu driven program in C was executed and the output was obtained.

L

FLOWCHART:



C PROGRAMMING EXERCISES**Scientific problem using decision making and looping****EX.NO: 5.1 GENERATING PRIME NUMBERS BETWEEN A GIVEN RANGES****DATE :****AIM:**

To write a C Program to generate prime numbers between two intervals.

ALGORITHM:

- Step 1: Start
- Step2: Declare variables
- Step 3: Get two intervals LOWER, UPPER
- Step 4: Set a FOR LOOP that traces the iteration till the upper limit.
- Step 4.1: Initialize the counter value as $n=LOWER+1$, condition as $n<UPPER$. Assign the flag value as $PRIME=1$.
- Step 5: Set a FOR LOOP to check whether the number is prime or not.
- Step 5.1.a: Initialize the counter value as $i=2$, condition as $i<n$.
- Step 5.1.b: Extract the remainder of number 'n' by dividing with counter value 'i'.
- Step 5.1.c: If the remainder is 0 then set the flag PRIME value to 0 and go to Step 4.1, Increment the counter value by 1 as 'n++'.
- Step 5.1.d: Else, Increment the counter value by 1 and do the step 5 to 5.1.c.
- Step 5.2: Check the flag value.
- Step 5.2.a: If the flag value is 1 then print the number as PRIME.
- Step 6: Stop

L

OUTPUT:

```
C:\ D:\WINDOWS\system32\cmd.exe - tc
-----
PROGRAM TO FIND THE PRIME NUMBERS BETWEEN A GIVEN RANGE
-----
ENTER THE LOWER LIMIT...
5
ENTER THE UPPER LIMIT...:
20
PRIME NUMBERS ARE...:
7
11
13
17
19
-----
```



PROGRAM:

```

#include<stdio.h>
void main()
{
    int i, prime, up, low, n;
    clrscr();
    printf("\n-----");
    printf("\n\nPROGRAM TO FIND THE PRIME NUMBERS BETWEEN A GIVEN RANGE");
    printf("\n\n-----");
    printf("\n\n\t ENTER THE LOWER LIMIT...: ");
    scanf("%d",&low);
    printf("\n\n\t ENTER THE UPPER LIMIT...: ");
    scanf("%d",&up);
    printf("\n\n\t PRIME NUMBERS ARE...: ");
    for(n=low+1; n<up; n++)
    {
        prime = 1;
        for(i=2; i<n; i++)
        {
            if(n%i == 0)
            {
                prime = 0;
                break;
            }
        }
        if(prime)
            printf("\n\n\t\t\t%d",n);
    }
    printf("\n\n-----");
    getch();
}

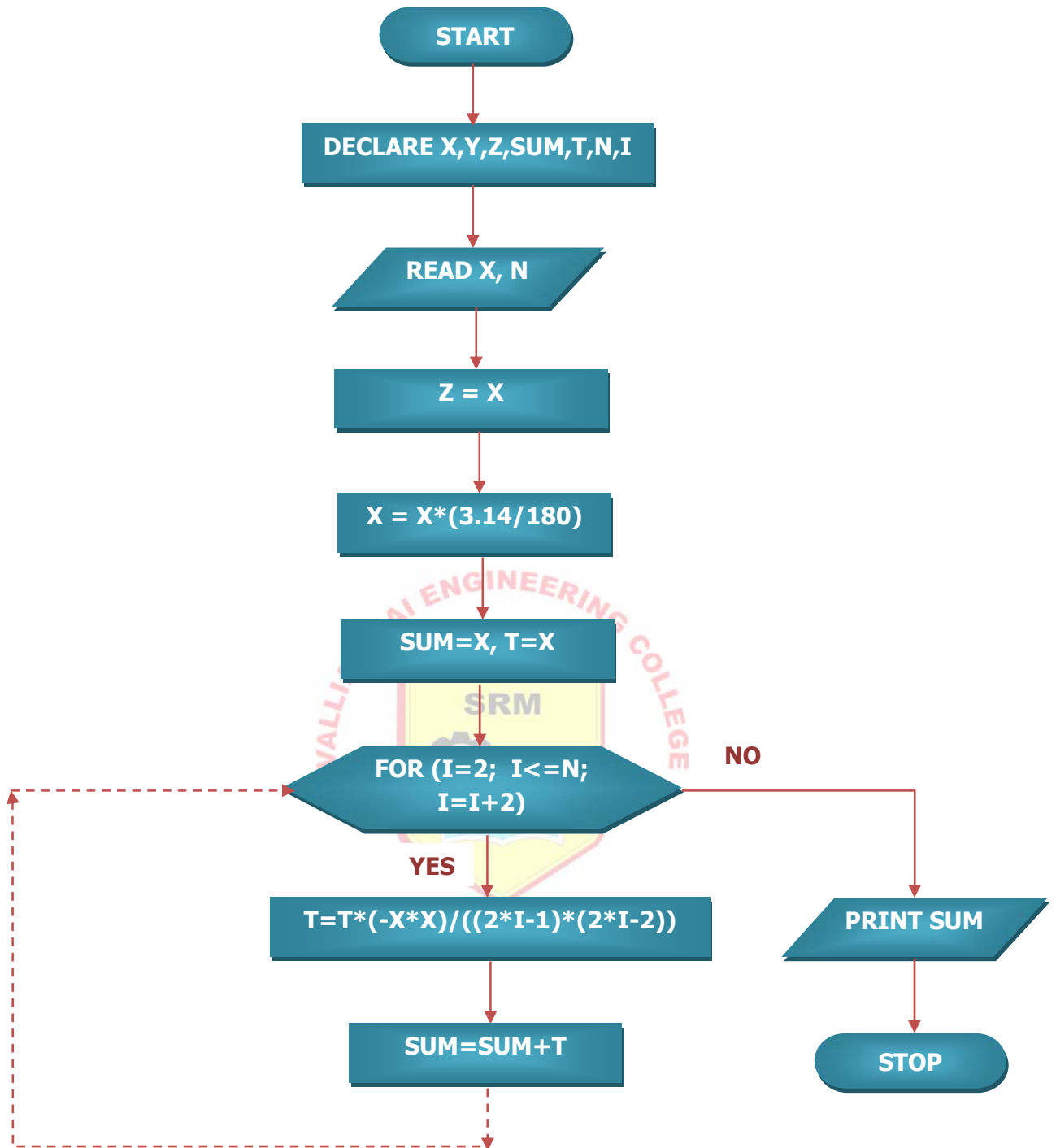
```

RESULT:

Thus a C program to generate prime numbers between two intervals was executed and the output was obtained.

L

FLOWCHART:



EX.NO : 5.2**GENERATING SINE SERIES****DATE :****AIM:**

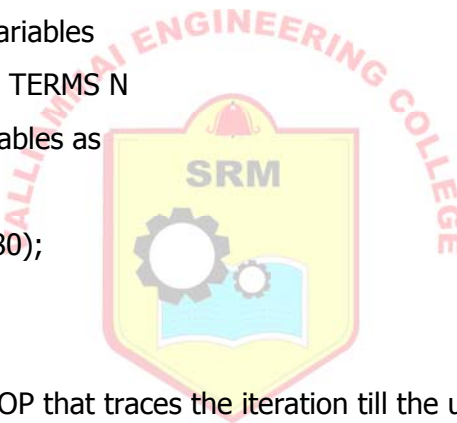
To write a C Program to print the sine series.

Formula of Sine Series:

$$\sin x = \sum_{n=0}^{\infty} \frac{(-1)^n}{(2n+1)!} x^{2n+1} = x - \frac{x^3}{3!} + \frac{x^5}{5!} - \dots \text{ for all } x$$

ALGORITHM:

- Step 1: Start
- Step2: Declare the variables
- Step3: Get ANGLE X, TERMS N
- Step4: Calculate variables as
- Z=X;
- X=X*(3.14/180);
- SUM=X;
- T=X;
- Step5: Set a FOR LOOP that traces the iteration till the upper limit.
- Step 5.1: Initialize the counter value as i=2, condition as I<=N
- Step5.2: Calculate T=T*(-X*X)/((2*I-1)*(2*I-2))
- Step 5.3: SUM=SUM+T;
- Step 5.4: Increment the counter value by 2 as 'I+2'. Check the condition I<=N, If it is true then go to step 5.2.
- Step 5.5: Else, PRINT SUM
- Step 6: Stop



L

OUTPUT:

```
D:\WINDOWS\system32\cmd.exe - tc

PROGRAM TO PRINT SINE SERIES
-----
ENTER THE ANGLE: 45
ENTER THE NUMBER OF TERMS: 3
THE VALUE OF SIN (45) IS 0.70

PROGRAM TO PRINT SINE SERIES
-----
ENTER THE ANGLE: 30
ENTER THE NUMBER OF TERMS: 20
THE VALUE OF SIN (30) IS 0.50
```



PROGRAM:

```

#include<stdio.h>
#include<math.h>
#include<conio.h>
void main()
{
    int i,n;
    float x,y,z,sum,t;
    clrscr();
    printf("\t\t PROGRAM TO PRINT SINE SERIES\t\t\n");
    printf("\t\t -----t\t\n");
    printf("\n ENTER THE ANGLE:");
    scanf("%f", &x);
    printf("\n ENTER THE NUMBER OF TERMS:");
    scanf("%d", &n);

    z=x;
    x=x*(3.14/180);
    sum=x;
    t=x;

    for(i=2; i<=n; i=i+2)
    {
        t=t*(-x*x)/((2*i-1)*(2*i-2));
        sum=sum+t;
    }
    printf("\n\n THE VALUE OF SIN(%5.2f) IS %5.2f\n", z, sum);
    getch();
}

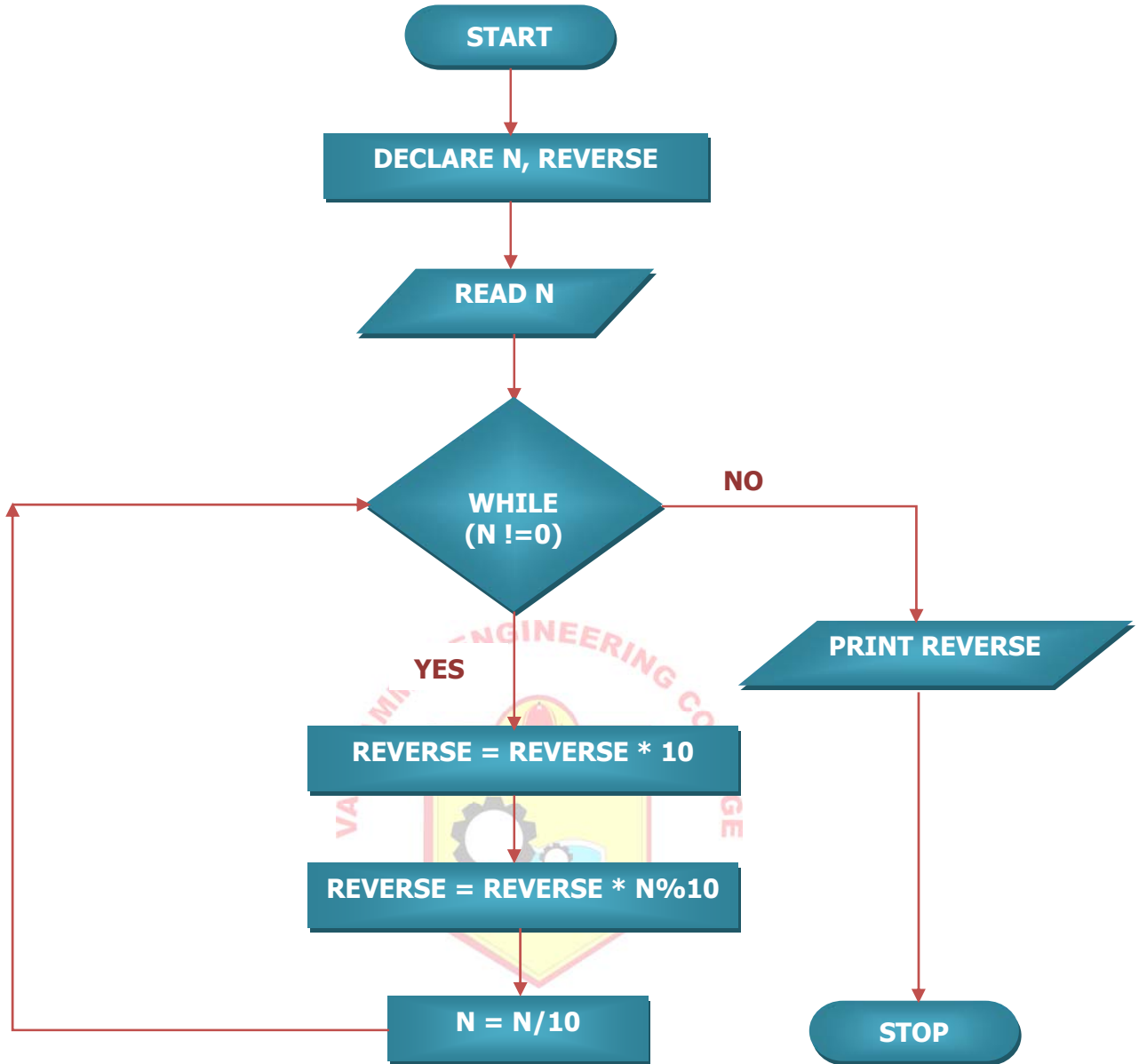
```

**RESULT:**

Thus a C program to print the sine series was executed and the output was obtained.

L

FLOWCHART:



EX.NO: 5.3**REVERSING THE DIGITS OF A NUMBER****DATE :****AIM:**

To write a C Program to reverse the digits of a number.

ALGORITHM:

- Step 1: Start
- Step2: Declare the variables
- Step 3: Read input N
- Step 4: Set a loop using WHILE until $N \neq 0$. If the condition is true then
- Step 4.1: REVERSE = REVERSE * 10;
- Step 4.2: REVERSE = REVERSE + $N\%10$;
- Step 4.3: $N = N/10$;
- Step 5: If condition fails, then PRINT REVERSE
- Step 6: Stop



L

OUTPUT:

```
D:\WINDOWS\system32\cmd.exe - tc  
  
REVERSAL OF A GIVEN NUMBER  
-----  
ENTER A NUMBER TO REVERSE:  
1234  
REVERSE OF ENTERED NUMBER IS: 4321
```



PROGRAM:

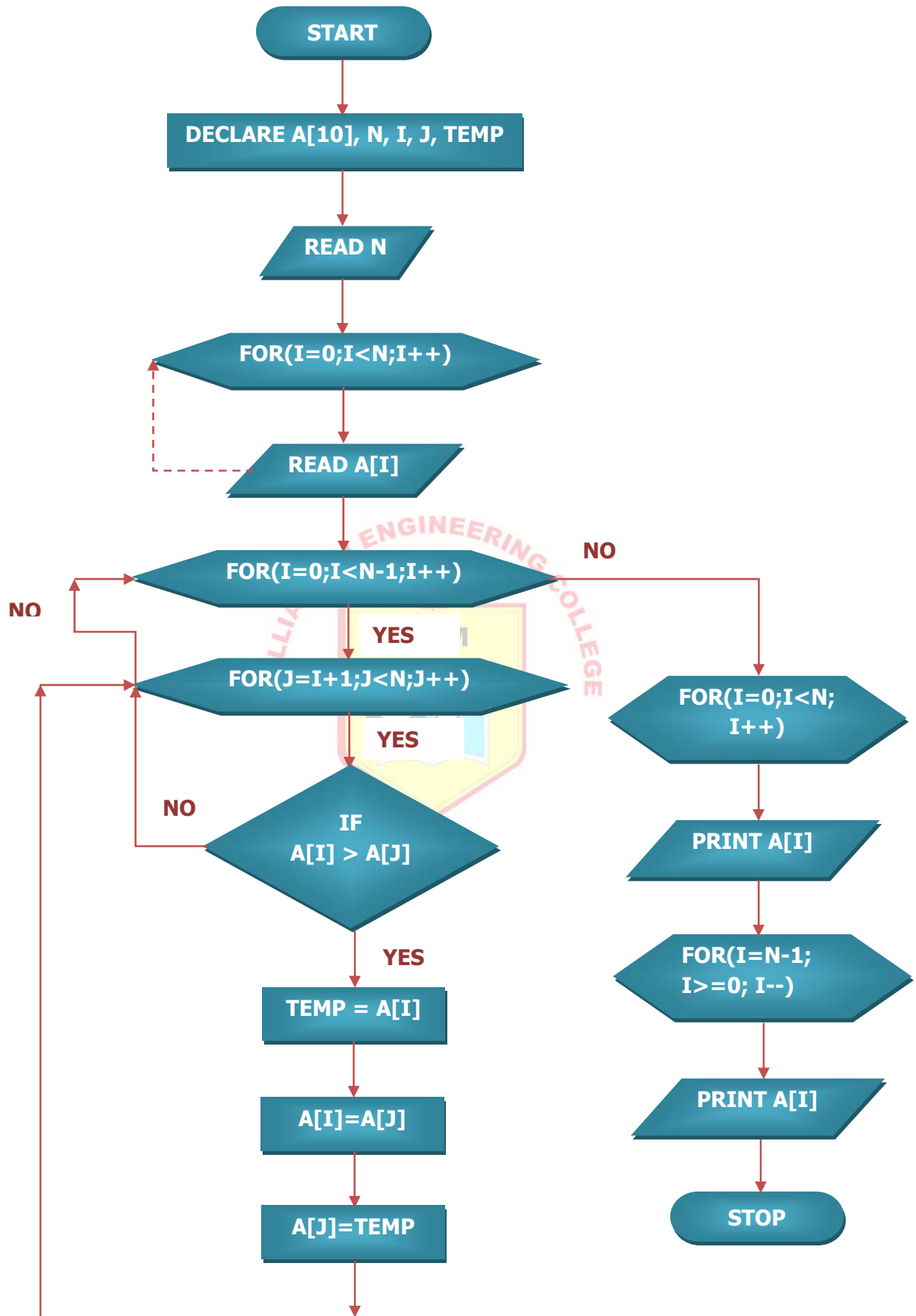
```
#include<stdio.h>
#include<conio.h>
void main()
{
    int n, reverse = 0;
    clrscr();
    printf("REVERSAL OF A GIVEN NUMBER\n");
    printf("-----\n");
    printf("ENTER A NUMBER TO REVERSE: \n");
    scanf("%d",&n);
    while (n != 0)
    {
        reverse = reverse * 10;
        reverse = reverse + n%10;
        n = n/10;
    }
    printf("REVERSE OF ENTERED NUMBER IS: %d\n", reverse);
    getch();
}
```

**RESULT:**

Thus a C program to reverse the digits of a number was executed and the output was obtained.

L

FLOWCHART:



C PROGRAMMING EXERCISES**Simple programming for one dimensional and two dimensional arrays****EX.NO: 6.1 ASCENDING AND DESCENDING ORDER OF THE GIVEN NUMBERS****DATE :****AIM:**

To write a C Program to print numbers in Ascending and Descending order.

ALGORITHM:

- Step 1: Start
- Step 2: Declare variables
- Step 3: Get the no. of terms N.
- Step 4: Get the numbers using for loop and store it in array A[10].
- Step 5: Set two for loops to arrange the given numbers in ascending order in the array A[10].
- Step 5.1: Check IF(A[I]>A[j]) then exchange the current two numbers and continue the same process until end of the array A[10].
- Step 6: Print the numbers in Ascending order and Descending order using for loop.
- Step 7: Stop

L

OUTPUT:

```
D:\WINDOWS\system32\cmd.exe - tc

ASCENDING AND DESCENDING ORDER OF THE GIVEN NUMBER
-----
Enter the no. of terms: 5
Enter the numbers:
34
85
22
47
17

Ascending order : 17 22 34 47 85
Descending order: 85 47 34 22 17
```



PROGRAM:

```

#include<stdio.h>
void main()
{
    int a[10],n,i,j,temp;
    clrscr();
    printf("\nASCENDING AND DESCENDING ORDER OF THE GIVEN NUMBERS");
    printf("\n-----");
    printf("\nEnter the no. of terms :");
    scanf("%d",&n);
    printf("\nEnter the numbers:\n");
    for(i=0;i<n;i++)
    {
        scanf("%d",&a[i]);
    }
    for(i=0;i<n-1;i++)
    {
        for(j=i+1;j<n;j++)
        {
            if(a[i]>a[j])
            {
                temp=a[i];
                a[i]=a[j];
                a[j]=temp;
            }
        }
    }
    printf("\nAscending order :");
    for(i=0;i<n;i++)
    printf("%d\t",a[i]);
    printf("\nDescending order:");
    for(i=n-1;i>=0;i--)
    printf("%d\t",a[i]);
    getch();
}

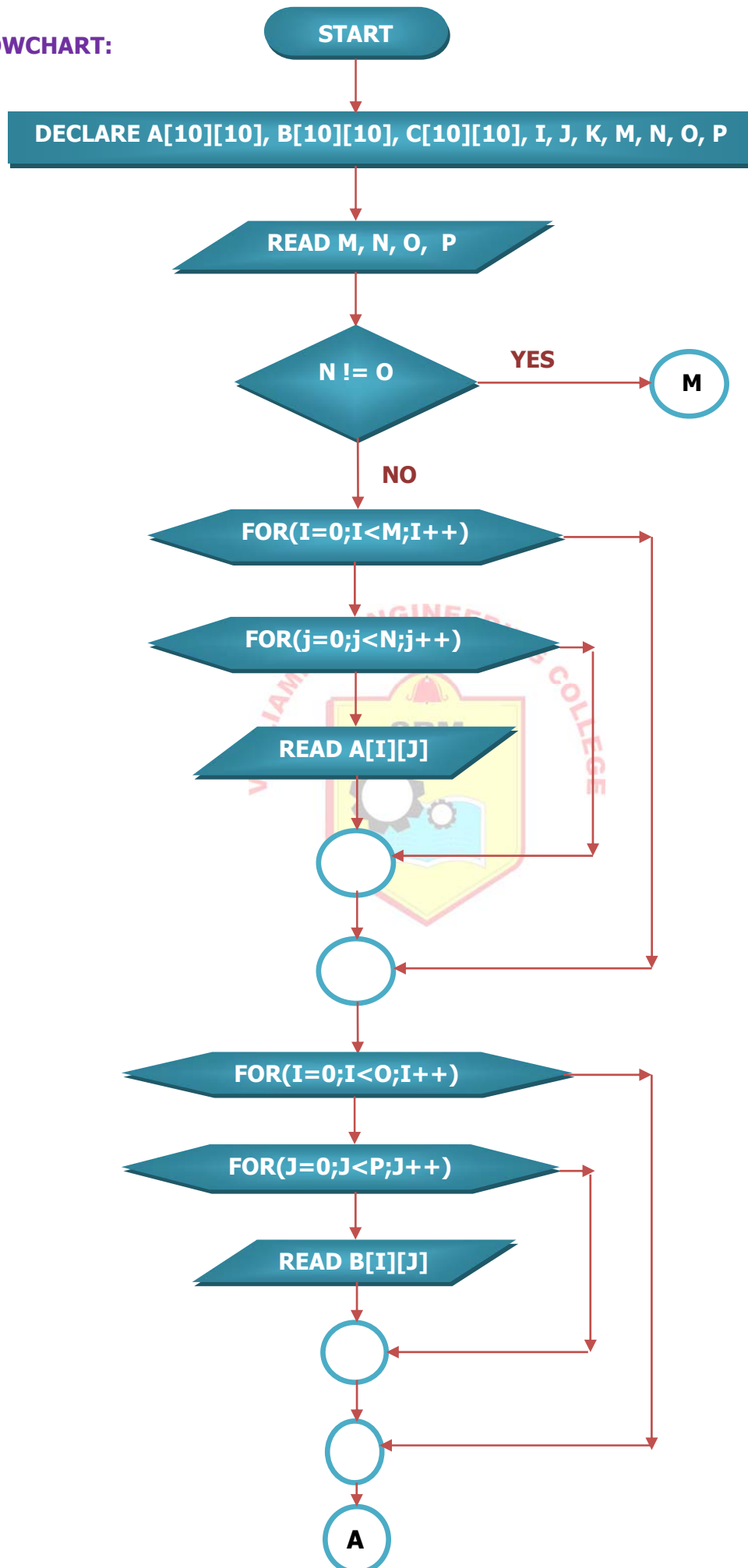
```

**RESULT:**

Thus a C program to print numbers in Ascending and Descending order was executed and the output was obtained.

L

FLOWCHART:

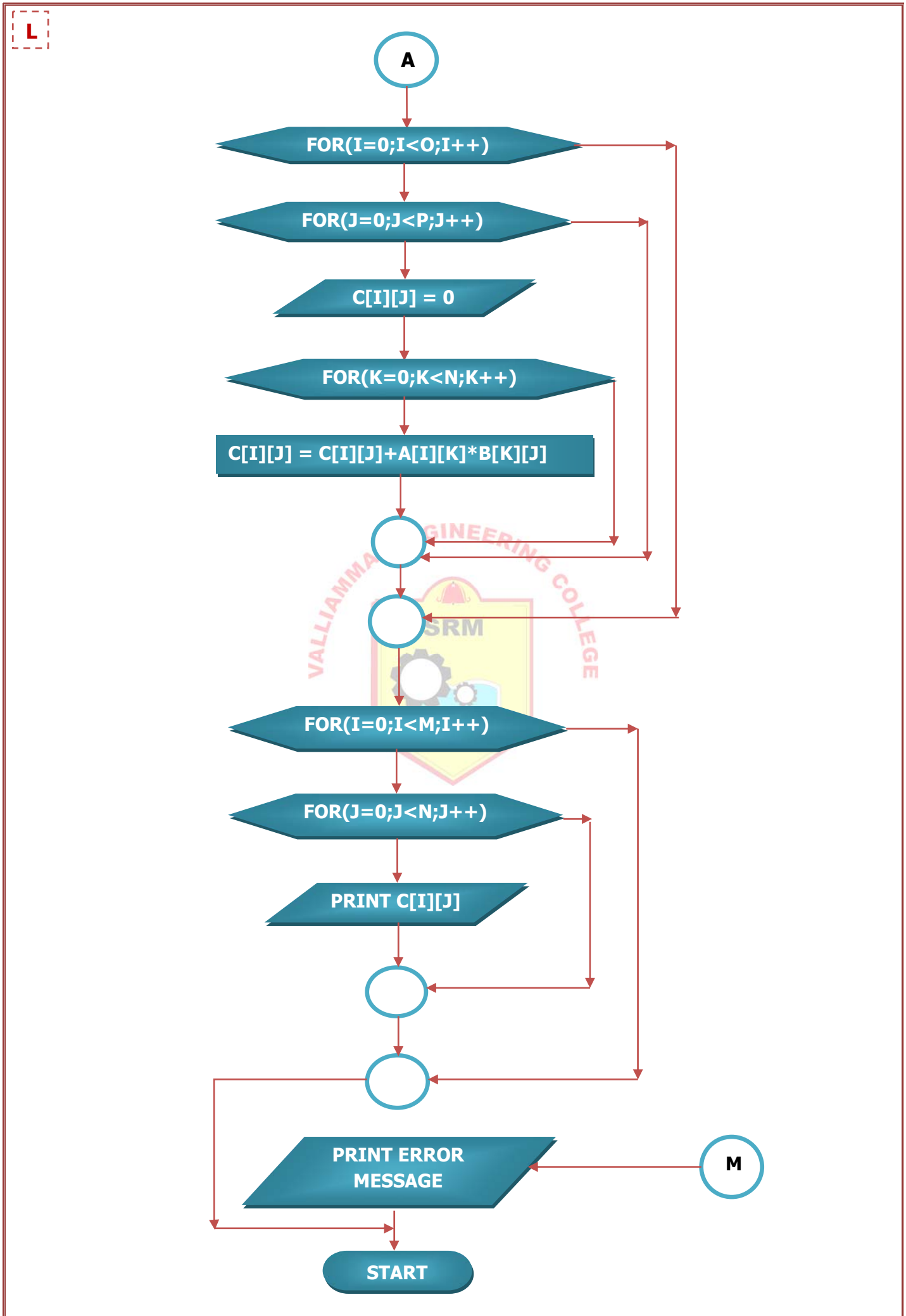


EX.NO: 6.2**MATRIX MULTIPLICATION****DATE :****AIM:**

To write a C Program to perform Matrix multiplication using array.

ALGORITHM:

- Step 1: Start
- Step 2: Declare variables
- Step 3: Get the rows and columns of two matrices M, N, O, P respectively.
- Step 4: Check N is not equal to O, if go to step 10.
- Step 5: Set a loop and get the elements of first matrix $A[i][j]$.
- Step 6: Set a loop and get the elements of second matrix $B[i][j]$.
- Step 7: Repeat the step 6 until $i < m$, $j < p$ and $k < n$.
- Step 8: Initialise $C[i][j]=0$ and multiply the two matrices and store the resultant in $C[i][j]= C[i][j]+A[i][k]*B[k][j]$.
- Step 9: Print the resultant matrix $C[i][j]$ and go to step 11.
- Step 10: Print the message "Column of first matrix must be same as row of second matrix".
- Step 11: Stop

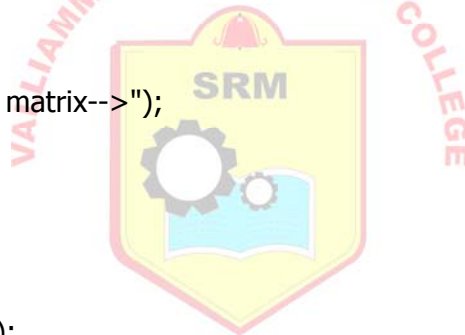


PROGRAM:

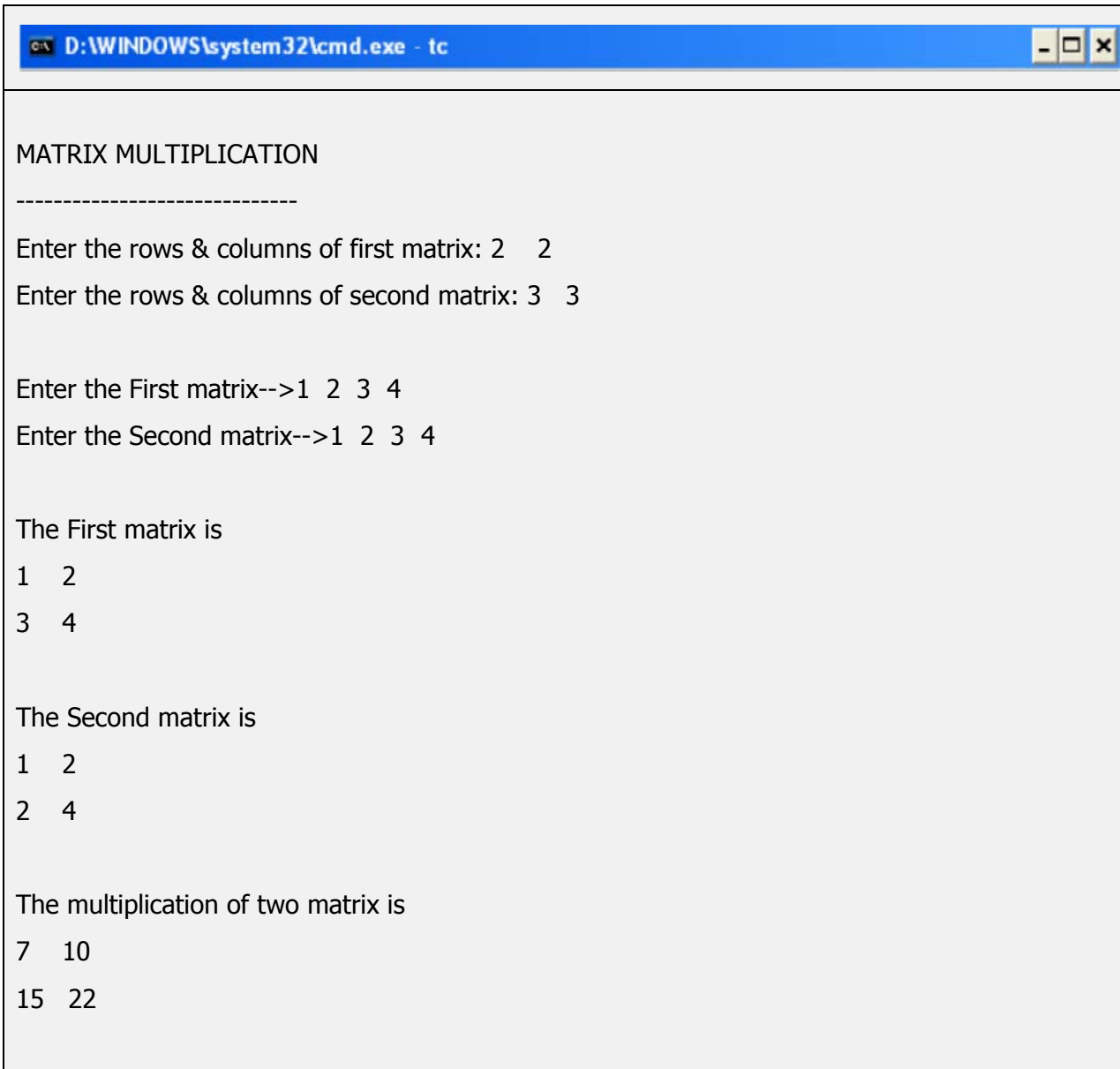
```

#include<stdio.h>
void main()
{
int a[10][10],b[10][10],c[10][10],i,j,k,m,n,o,p;
clrscr();
printf("\nMATRIX MULTIPLICATION\n");
printf("\n-----\n");
printf("\nEnter the rows & columns of first matrix: ");
scanf("%d %d",&m,&n);
printf("\nEnter the rows & columns of second matrix: ");
scanf("%d %d",&o,&p);
if(n!=o)
{
printf("Matrix mutiplication is not possible");
printf("\nColumn of first matrix must be same as row of second matrix");
}
else
{
printf("\nEnter the First matrix-->");
for(i=0;i<m;i++)
{
for(j=0;j<n;j++)
{
scanf("%d",&a[i][j]);
}
}
printf("\nEnter the Second matrix-->");
for(i=0;i<o;i++)
{
for(j=0;j<p;j++)
{
scanf("%d",&b[i][j]);
}
}
printf("\n\nThe First matrix is\n");
for(i=0;i<m;i++)
{
printf("\n");
for(j=0;j<n;j++)

```



L

OUTPUT:

```
ca D:\WINDOWS\system32\cmd.exe - tc

MATRIX MULTIPLICATION
-----
Enter the rows & columns of first matrix: 2 2
Enter the rows & columns of second matrix: 3 3

Enter the First matrix-->1 2 3 4
Enter the Second matrix-->1 2 3 4

The First matrix is
1 2
3 4

The Second matrix is
1 2
2 4

The multiplication of two matrix is
7 10
15 22
```

```

    {
        printf("%d\t",a[i][j]);
    }
}
printf("\n\nThe Second matrix is\n");
for(i=0;i<o;i++)
{
    printf("\n");
    for(j=0;j<p;j++)
    {
        printf("%d\t",b[i][j]);
    }
}
for(i=0;i<m;i++) //row of first matrix
{
    for(j=0;j<p;j++) //column of second matrix
    {
        c[i][j]=0;
        for(k=0;k<n;k++)
        {
            c[i][j]= c[i][j]+a[i][k]*b[k][j];
        }
    }
}
printf("\n\nThe multiplication of two matrix is\n");
for(i=0;i<m;i++)
{
    printf("\n");
    for(j=0;j<p;j++)
    {
        printf("%d\t",c[i][j]);
    }
}
getch();
}

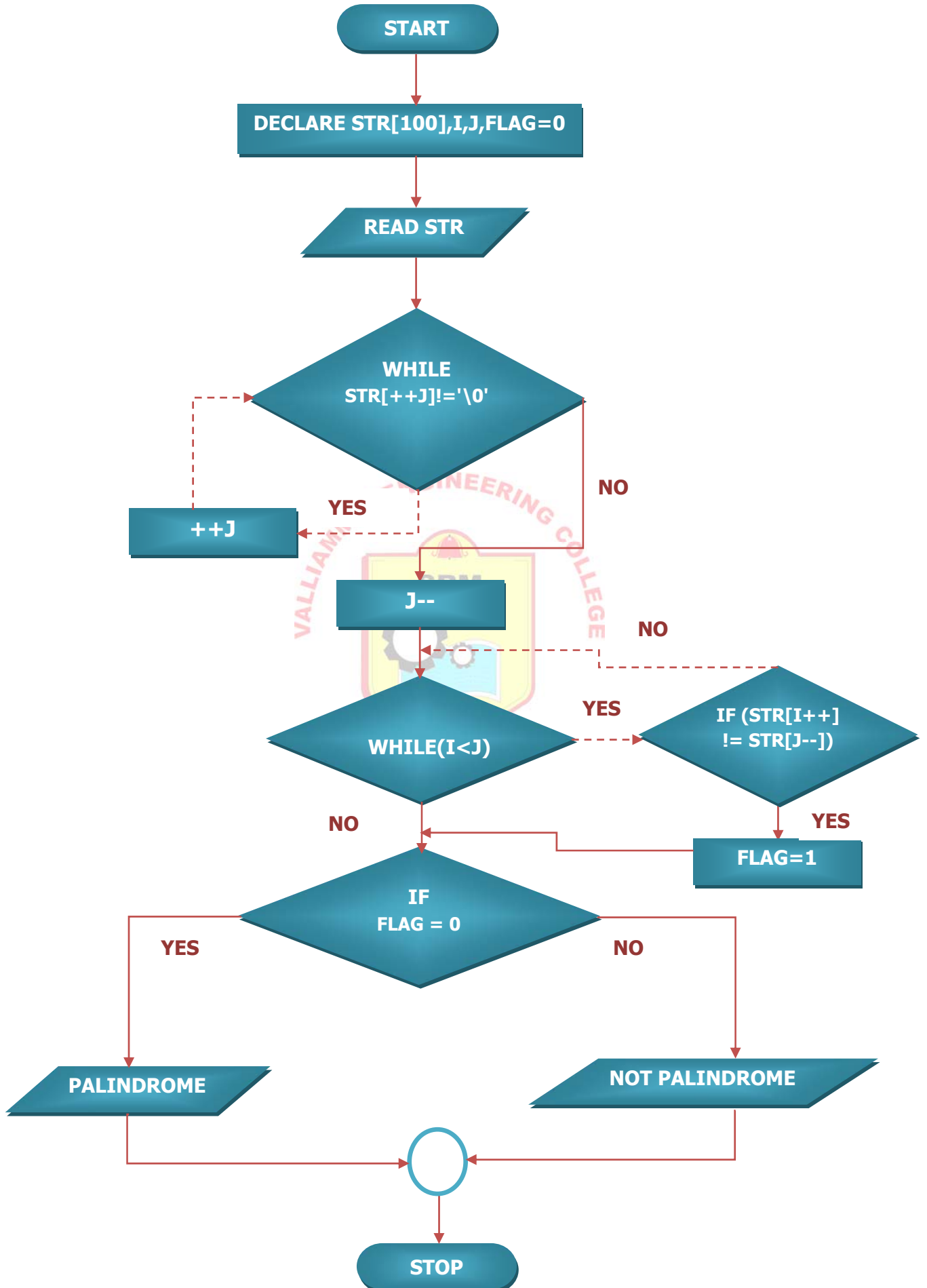
```

RESULT:

Thus a C program for the implementation of matrix multiplication was executed and the output was obtained.



FLOWCHART:



EX.NO: 6.3**STRING PALINDROME****DATE :****AIM:**

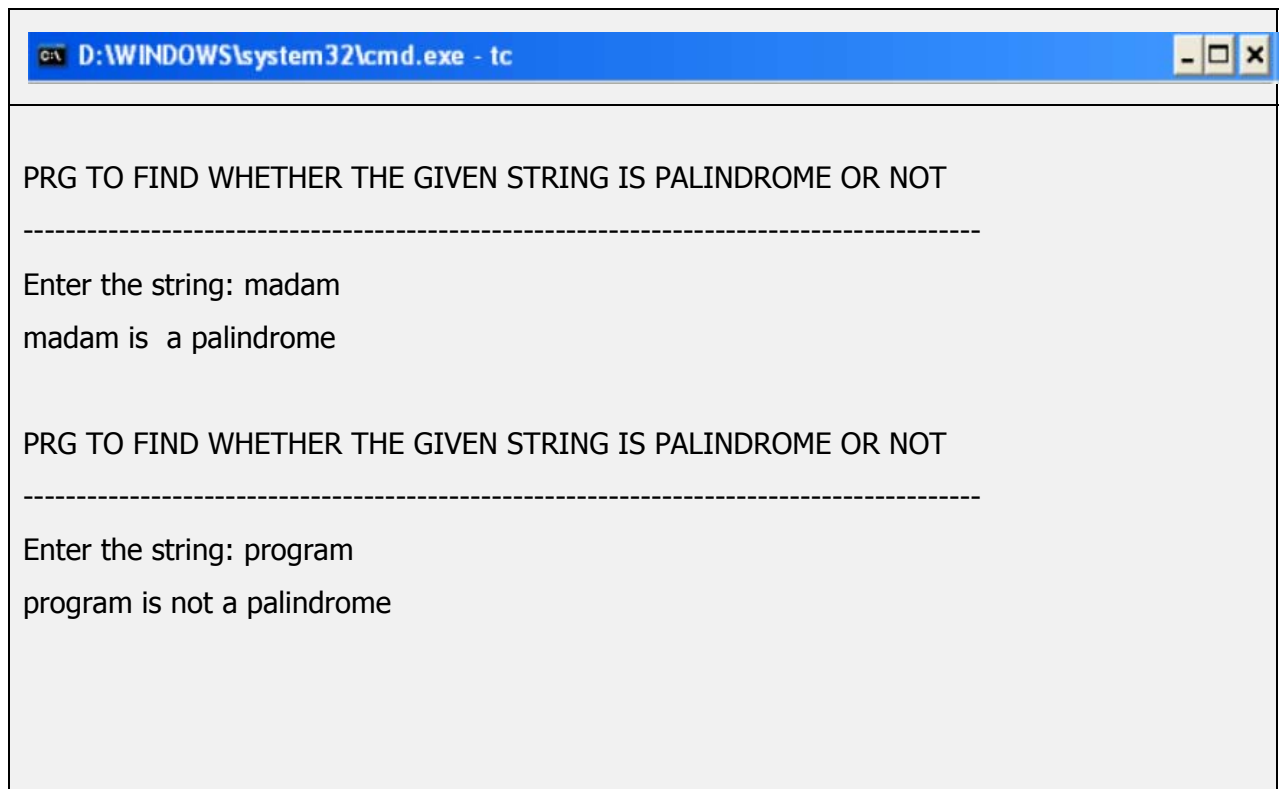
To write a C Program to find whether the given string is Palindrome or not without using string functions.

ALGORITHM:

- Step 1: Start
- Step 2: Declare variables and character array with necessary size.
- Step 3: Get the string STR
- Step 4: Set a while loop to count the character in the string. WHILE(STR[++J]!='\0')
- Step 5: Decrement the character position (i.e) to reach last position by J--;
- Step 6: Set a while loop to check the each character until it reaches the end of the string. If WHILE(I<J) is fails then go to Step 7.
- Step 6.1: Check IF(STR[I++] != STR[J--]), then set flag=1. Else, increment I and decrement J and go to step 6.
- Step 7: If two string are same i.e flag = 0 then print as "Palindrome" else print as "Not Palindrome".
- Step 8: Stop

L

OUTPUT:



```
ca D:\WINDOWS\system32\cmd.exe - tc
PRG TO FIND WHETHER THE GIVEN STRING IS PALINDROME OR NOT
-----
Enter the string: madam
madam is a palindrome

PRG TO FIND WHETHER THE GIVEN STRING IS PALINDROME OR NOT
-----
Enter the string: program
program is not a palindrome
```



PROGRAM:

```

#include<stdio.h>
void main()
{
    char str[100];
    int i=0,j=-1,flag=0;
    clrscr();
    printf("\nPRG TO FIND WHETHER THE GIVEN STRING IS PALINDROME OR NOT");
    printf("\n-----");
    printf("\n\nEnter a string: ");
    scanf("%s",str);
    while(str[++j]!='\0');
    j--;

    while(i<j)
    {
        if(str[i++] != str[j--])
        {
            flag=1;
            break;
        }
    }

    if(flag == 0)
        printf("\n\nThe string %s is a palindrome.", str);
    else
        printf("\n\nThe string %s is not a palindrome.", str);

    getch();
}

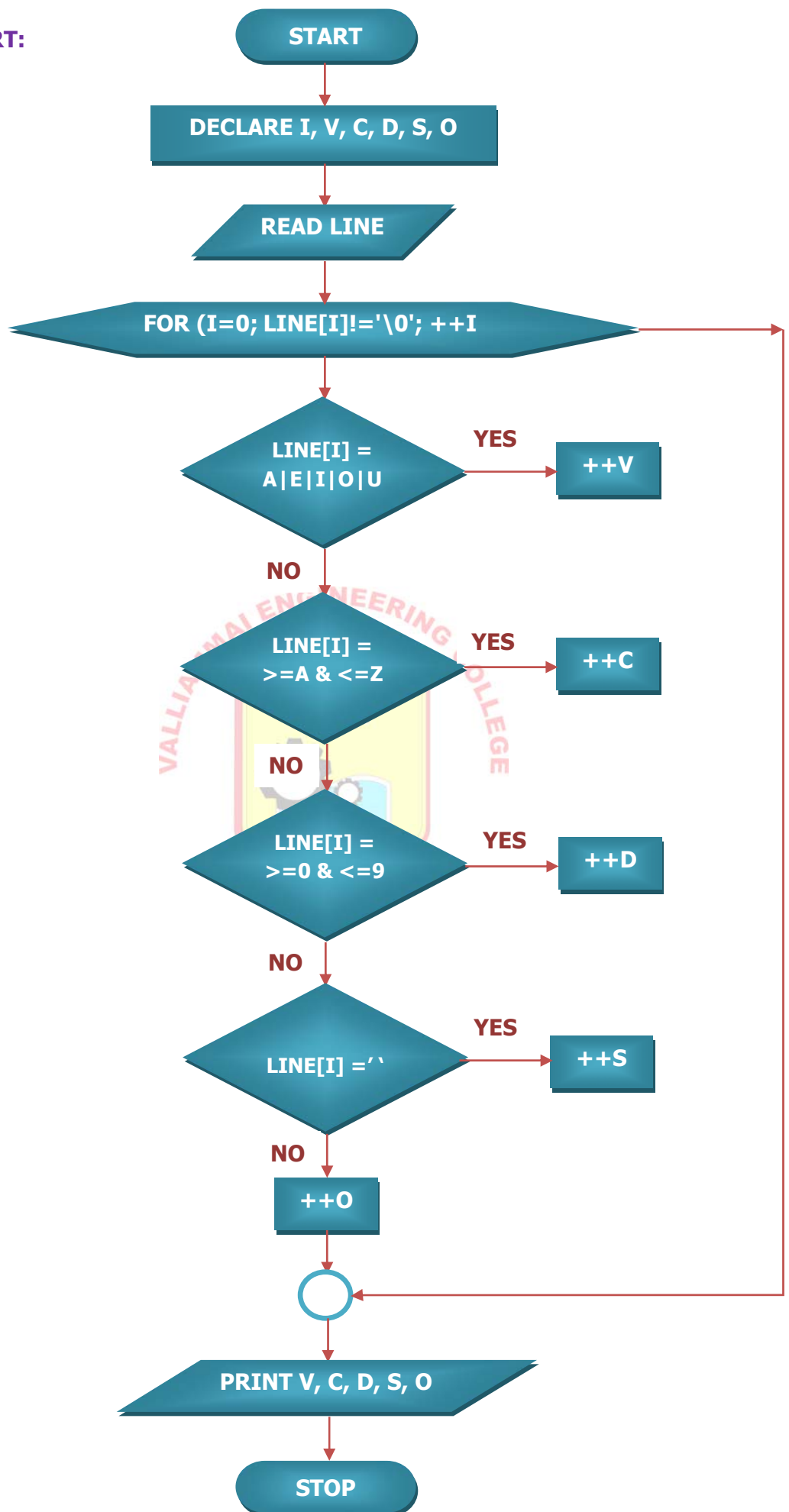
```

**RESULT:**

Thus a C Program to find whether the given string is Palindrome or not without using string functions was executed and the output was obtained.

L

FLOWCHART:



C PROGRAMMING EXERCISES**Solving problems using String functions****EX.NO: 7.1****STRING MANIPULATION****DATE :****AIM:**

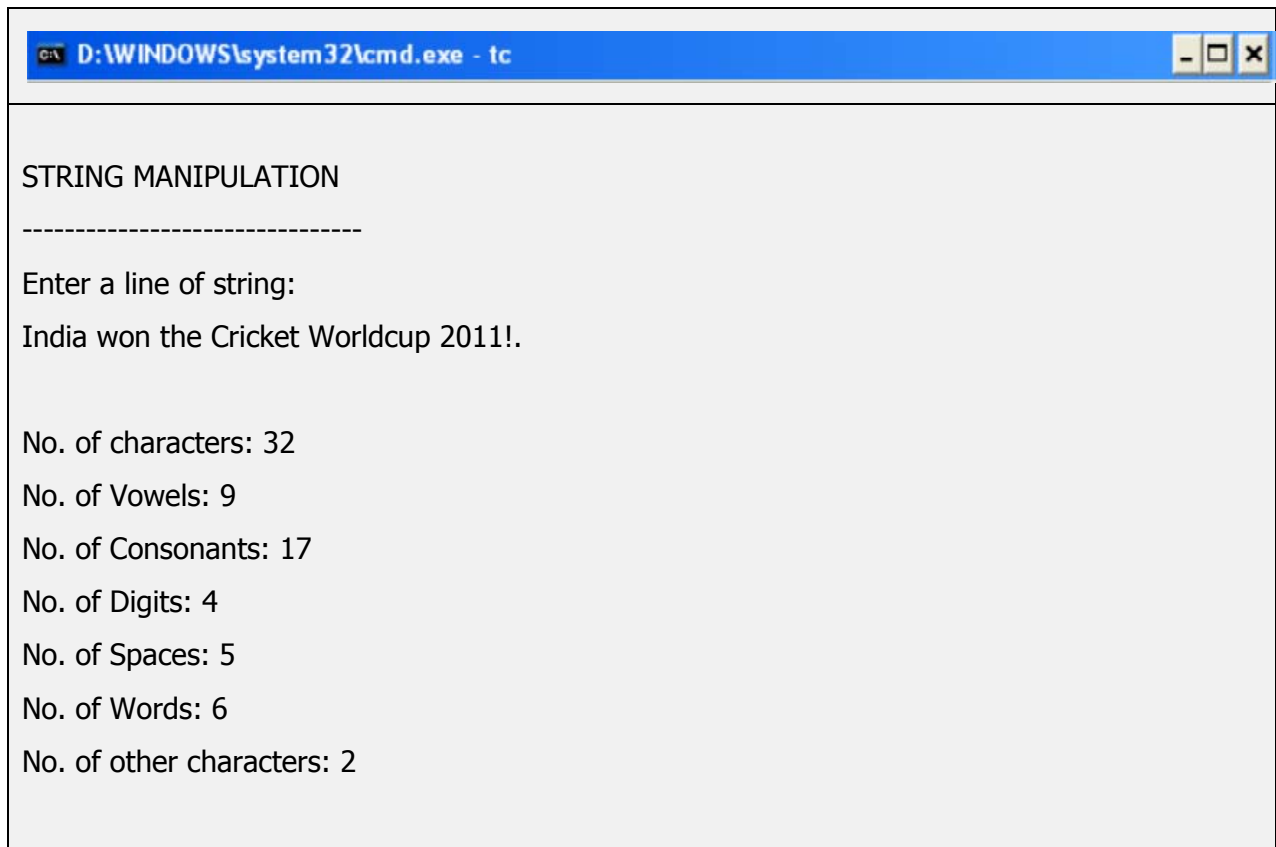
To write a C Program to count the number of characters, vowels, consonants, spaces, words & other characters using string functions.

ALGORITHM:

- Step 1: Start
- Step 2: Declare the variables V, C, D, S, O.
- Step 3: Get the string LINE using GETS() function.
- Step 4: Set a for loop to read each character of the given string until the occurrence of "\0". Initialize the counter value I=0.
- Step 4.1: If the character is a, e, i, o, u or A, E, I, O, U then increment the V by one. Go to step 4.
- Step 4.2: Else, if the character is between 'a' and 'z' or 'A' and 'Z' then increment C by one. Go to step 4.
- Step 4.3: Else, if the character is between 0 and 9 then increment the D by one. Go to step 4.
- Step 4.4: Else, if the character is whitespace ' ' then increment the S by one. Go to step 4.
- Step 4.5: Else, increment the O by one. Go to step 4.
- Step 5: Print V, C, D, S, O.
- Step 6: Stop

L

OUTPUT:



```
D:\WINDOWS\system32\cmd.exe - tc

STRING MANIPULATION
-----
Enter a line of string:
India won the Cricket Worldcup 2011!.

No. of characters: 32
No. of Vowels: 9
No. of Consonants: 17
No. of Digits: 4
No. of Spaces: 5
No. of Words: 6
No. of other characters: 2
```



PROGRAM:

```

#include<stdio.h>
#include<string.h>
void main(){
char line[150];
int i,v,c,d,s,o;
o=v=c=d=s=0;
clrscr();
printf("\n\t STRING MANIPULATION");
printf("\n\t-----");
printf("\n\n\t Enter a line of string:\n\n\t");
gets(line);
for(i=0;line[i]!='\0';++i)
{
    if(line[i]=='a' || line[i]=='e' || line[i]=='i' || line[i]=='o' || line[i]=='u' || line[i]=='A'
|| line[i]=='E' || line[i]=='I' || line[i]=='O' || line[i]=='U')
        ++v;
    else if((line[i]>='a' && line[i]<='z') || (line[i]>='A' && line[i]<='Z'))
        ++c;
    else if(line[i]>='0'&& line[i]<='9')
        ++d;
    else if (line[i]==' ')
        ++s;
    else ++o;
}
printf("\n\n\tNo. of Characters: %d",strlen(line)-s);
printf("\n\n\tNo. of Vowels: %d",v);
printf("\n\n\tNo. of Consonants: %d",c);
printf("\n\n\tNo. of Digits: %d",d);
printf("\n\n\tNo. of Spaces: %d",s);
printf("\n\n\tNo. of Words: %d",s+1);
printf("\n\n\tNo. of other characters: %d",o);
getch();
}

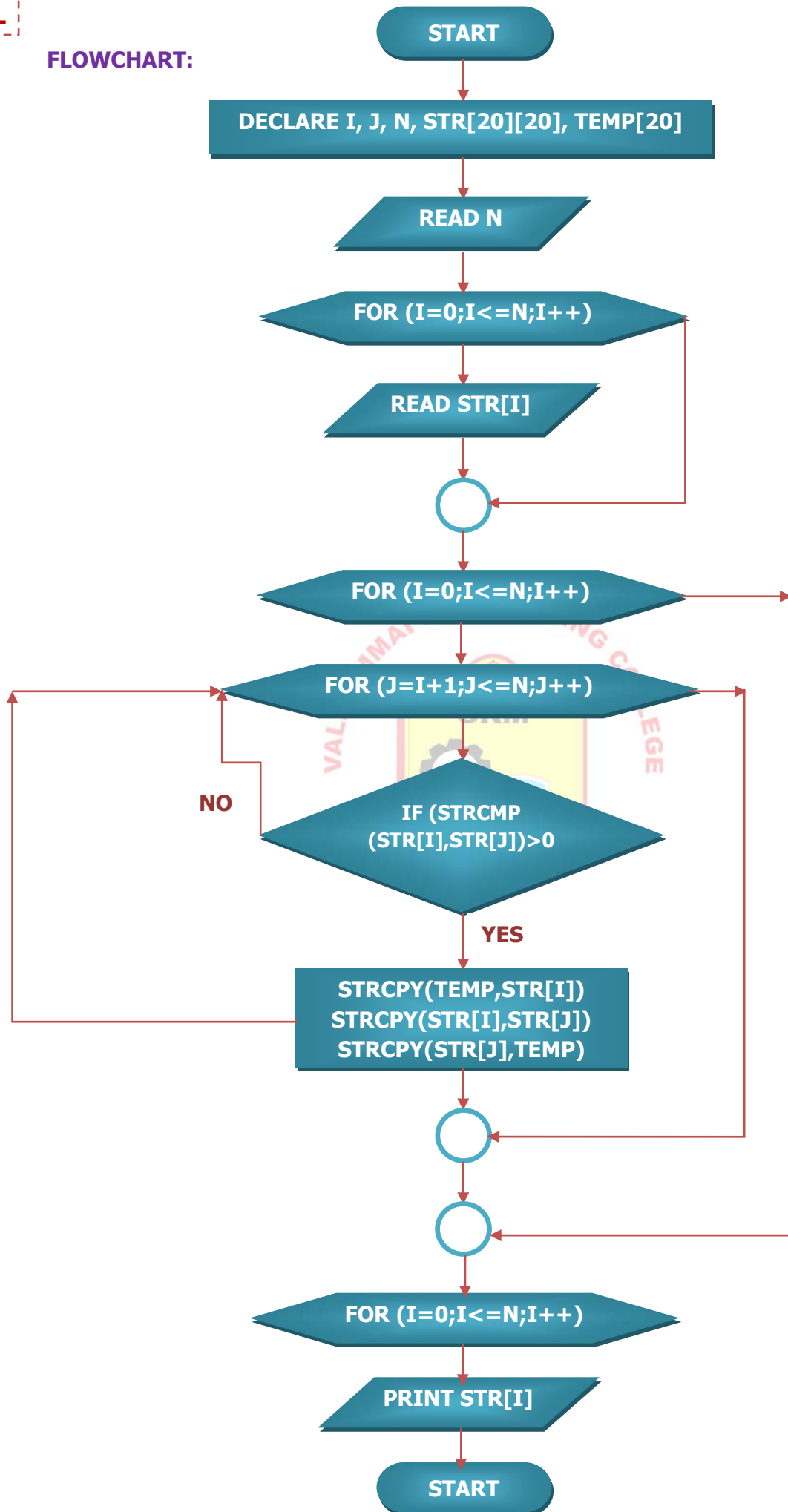
```

RESULT:

Thus a C Program to count the number of characters, vowels, consonants, spaces, words & other characters was executed and the output was obtained.

L

FLOWCHART:



EX.NO : 7.2**SORTING STRINGS****DATE :****AIM:**

To write a C program to sort strings using string functions.

ALGORITHM:

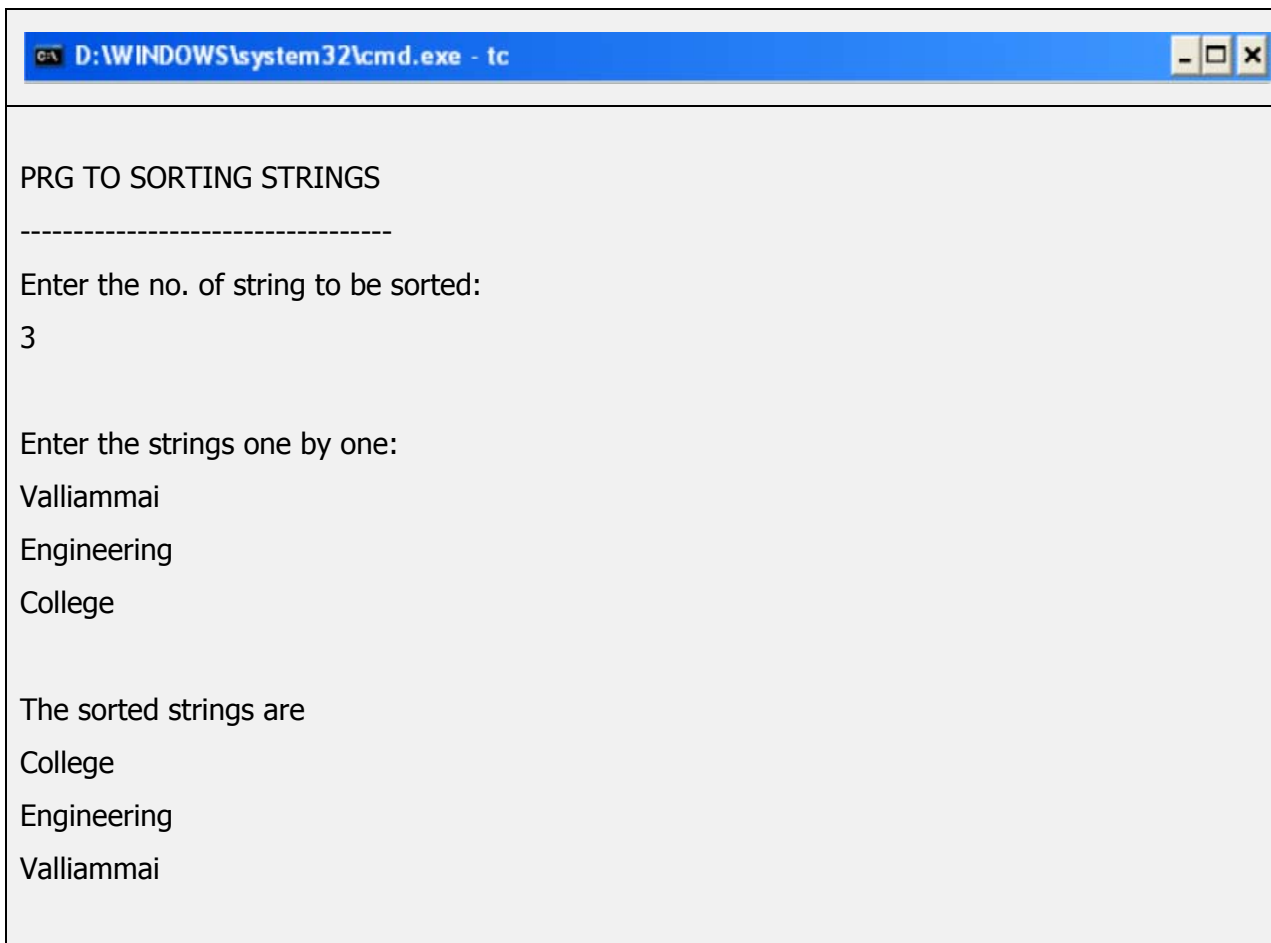
- Step 1: Start
- Step 2: Declare the variables I, J, N, STR[20][20], TEMP[20].
- Step 3: Get the no. of string N.
- Step 4: Get the string one by one by using GETS function with help of for loop.
- Step 5: Read the strings one by one with the help of two for loops and compare each string with other strings using STRCMP function. Repeat this until end of N.
- Step 6: If the two strings are different then exchange the strings using STRCPY function and go to step 5.
- Step 7: Finally, print the sorted strings using PUTS function.
- Step 8: Stop.

Note:

- **strcmp(string1, string2):** The string comparison starts with the first character in each string and continues with subsequent characters until the corresponding characters differ or until the end of the strings is reached.
- **strcpy(destination, source):** Copies string source to destination, stopping after the terminating null character has been moved.

L

OUTPUT:



```
D:\WINDOWS\system32\cmd.exe - tc

PRG TO SORTING STRINGS
-----
Enter the no. of string to be sorted:
3
Enter the strings one by one:
Valliammai
Engineering
College

The sorted strings are
College
Engineering
Valliammai
```



PROGRAM:

```

#include<stdio.h>
#include<string.h>
void main()
{
    int i,j,n;
    char str[20][20],temp[20];
    clrscr();
    puts("PRG TO SORT STRINGS\n");
    puts("-----\n\n");
    puts("Enter the no. of string to be sorted:\n");
    scanf("%d",&n);
    printf("\n\nEnter the strings one by one:\n\n");
    for(i=0;i<=n;i++)
    {
        gets(str[i]);
    }
    for(i=0;i<=n;i++)
    {
        for(j=i+1;j<=n;j++)
        {
            if(strcmp(str[i],str[j])>0)
            {
                strcpy(temp,str[i]);
                strcpy(str[i],str[j]);
                strcpy(str[j],temp);
            }
        }
    }
    printf("\n\nThe sorted strings are\n");
    for(i=0;i<=n;i++)
        puts(str[i]);
    getch();
}

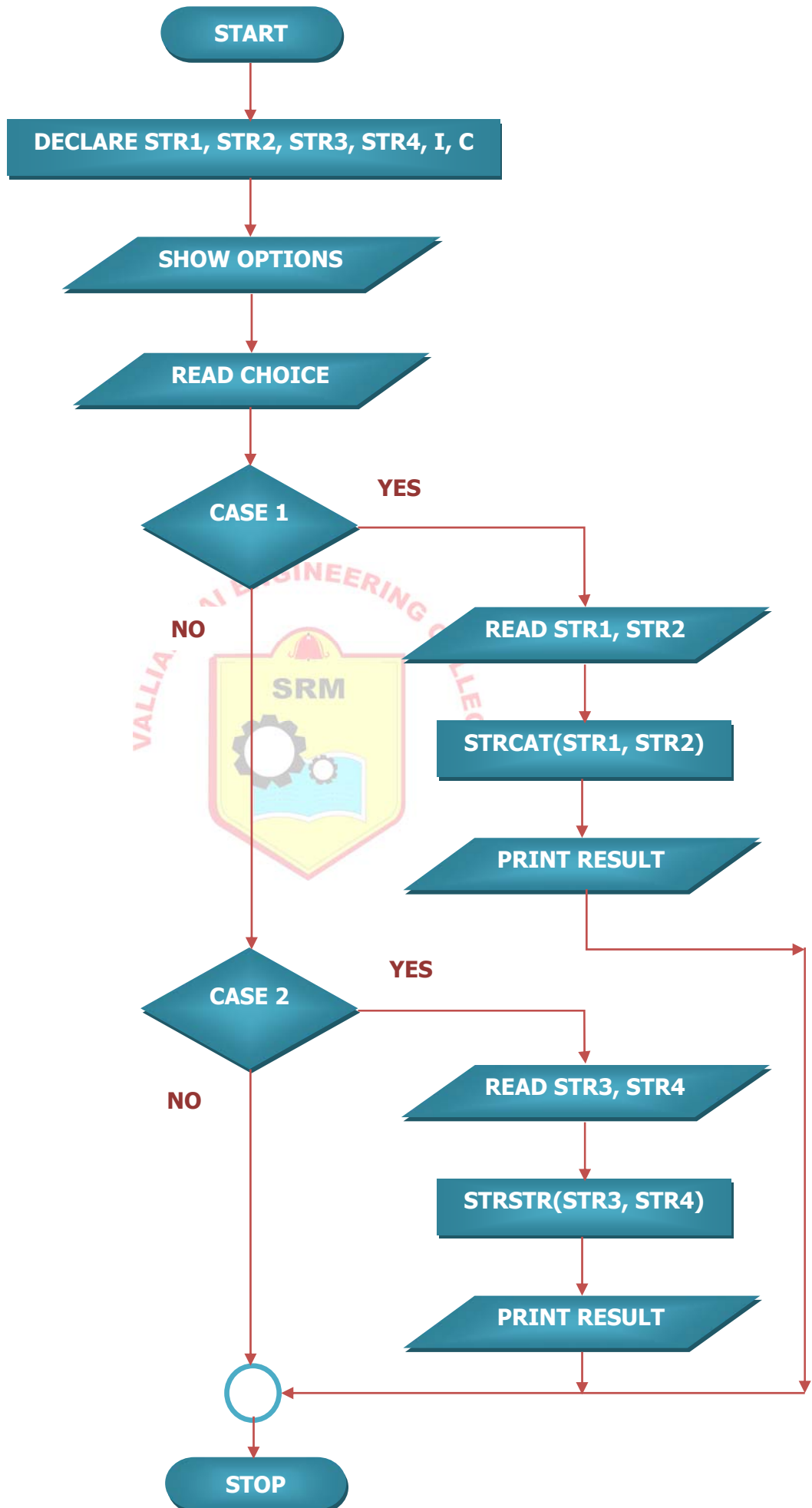
```

**RESULT:**

Thus a C Program to sort the given strings was executed and the output was obtained.



FLOWCHART:



EX.NO: 7.3**STRING OPERATIONS****DATE :****AIM:**

To write a C program to find the substring of a string & concatenation of two strings using string function.

ALGORITHM:

- Step 1: Start
- Step 2: Declare the variables.
- Step 3: Show the two options and get the choice as input from the user.
- Step 4: For concatenation of two strings, get the input strings from the user and use the string function STRCAT(STR1,STR2) and print the result.
- Step 5: For getting a substring from a string, get the string and substring to find from the user and use the string function STRSTR(STR3,STR4) and print the result.
- Step 6: Stop.

Note:

- **strcat(dest, src):** appends a copy of src string to the end of dest string.
- **strstr(string1, string2):** Finds the first occurrence of a substring string2 in main string string1.

L

OUTPUT:

```
D:\WINDOWS\system32\cmd.exe - tc

STRING OPERATIONS
-----
1. CONCATENATION OF TWO STRINGS
2. FINDING THE SUBSTRING

Enter your choice : 1

Enter the first string : Tamil
Enter the second string : Nadu

The concatenated string is TamilNadu

Enter your choice : 2

Enter a string : ValliammaiEnggCollege
Enter substring to find : Engg

The substring is EnggCollege
```

PROGRAM:

```

#include<stdio.h>
#include<string.h>
void main()
{
    char str1[30], str2[30], str3[30], str4[30];
    int i,c;
    clrscr();
    printf("\nSTRING OPERATIONS");
    printf("\n-----");
    printf("\n\n1.CONCATENATION OF TWO STRINGS\n");
    printf("\n2.FINDING THE SUBSTRING\n");
    printf("\n\nEnter your choice: ");
    scanf("%d",&c);
    switch(c)
    {
    case 1:
        printf("\n\nEnter the first string:");
        scanf("%s",str1);
        printf("\n\nEnter the second string:");
        scanf("%s",str2);
        printf("\n\nThe concatenated string is %s",strcat(str1,str2));
        break;
    case 2:
        printf("\n\nEnter a string:");
        scanf("%s",str3);
        printf("\n\nEnter substring to find:");
        scanf("%s",str4);
        printf("\n\nThe substring is: %s\n", strstr(str3, str4));
        break;
    default:
        printf("\n\nInvalid choice");
        break;
    }
    getch();
}

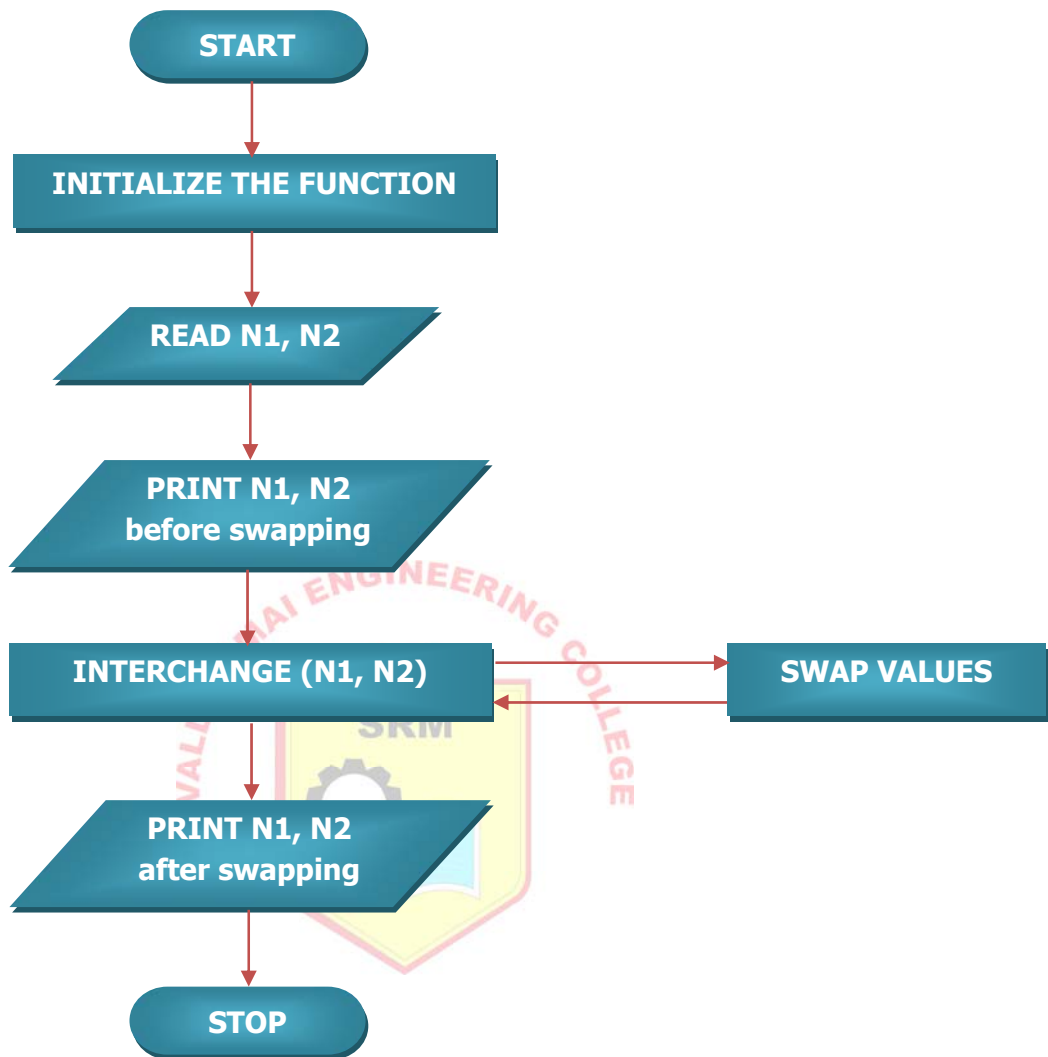
```

RESULT:

Thus a C Program to find the substring of a string & concatenation of two strings using string function was executed and the output was obtained.

L

FLOWCHART:



C PROGRAMMING EXERCISES**Programs with user defined functions – Includes Parameter Passing****EX.NO: 8.1 SWAPPING OF TWO NUMBERS - CALL BY VALUE****DATE :****AIM:**

To write a C Program to swap two values with call by value.

ALGORITHM:

- Step 1: Start
- Step 2: Initialize the function INTERCHANGE(N1,N2) & declare the variables N1, N2
- Step 3: Get the two numbers N1, N2
- Step 4: Print N1 and N2 values before swapping
- Step 5: Exchange the values by calling the function INTERCHANGE(N1,N2) by passing the values of N1 & N2.
- Step 6: Print N1 and N2 values after swapping
- Step 7: Stop

L

OUTPUT:

```
D:\WINDOWS\system32\cmd.exe - tc

SWAPPING OF TWO NUMBERS - CALL BY VALUE
-----
Enter the first number N1: 50
Enter the second number N2: 70

The N1 and N2 values before swapping: 50 70
The N1 and N2 values after swapping: 70 50
```



PROGRAM:

```

#include<stdio.h>
void interchange(int a, int b);
void main()
{
    int n1=0,n2=0;
    clrscr();
    printf("\nSWAPPING OF TWO NUMBERS - CALL BY VALUE");
    printf("\n-----");
    printf("\n\nEnter the first number N1: ");
    scanf("%d",&n1);
    printf("\n\nEnter the second number N2: ");
    scanf("%d",&n2);
    printf("\n\nThe N1 and N2 values before swapping: %d %d\n",n1,n2);
    interchange(n1,n2);
    getch();
}

void interchange(int a, int b)
{
    int temp;
    temp=a;
    a=b;
    b=temp;
    printf("\n\nThe N1 and N2 values after swapping: %d %d\n",a,b);
}

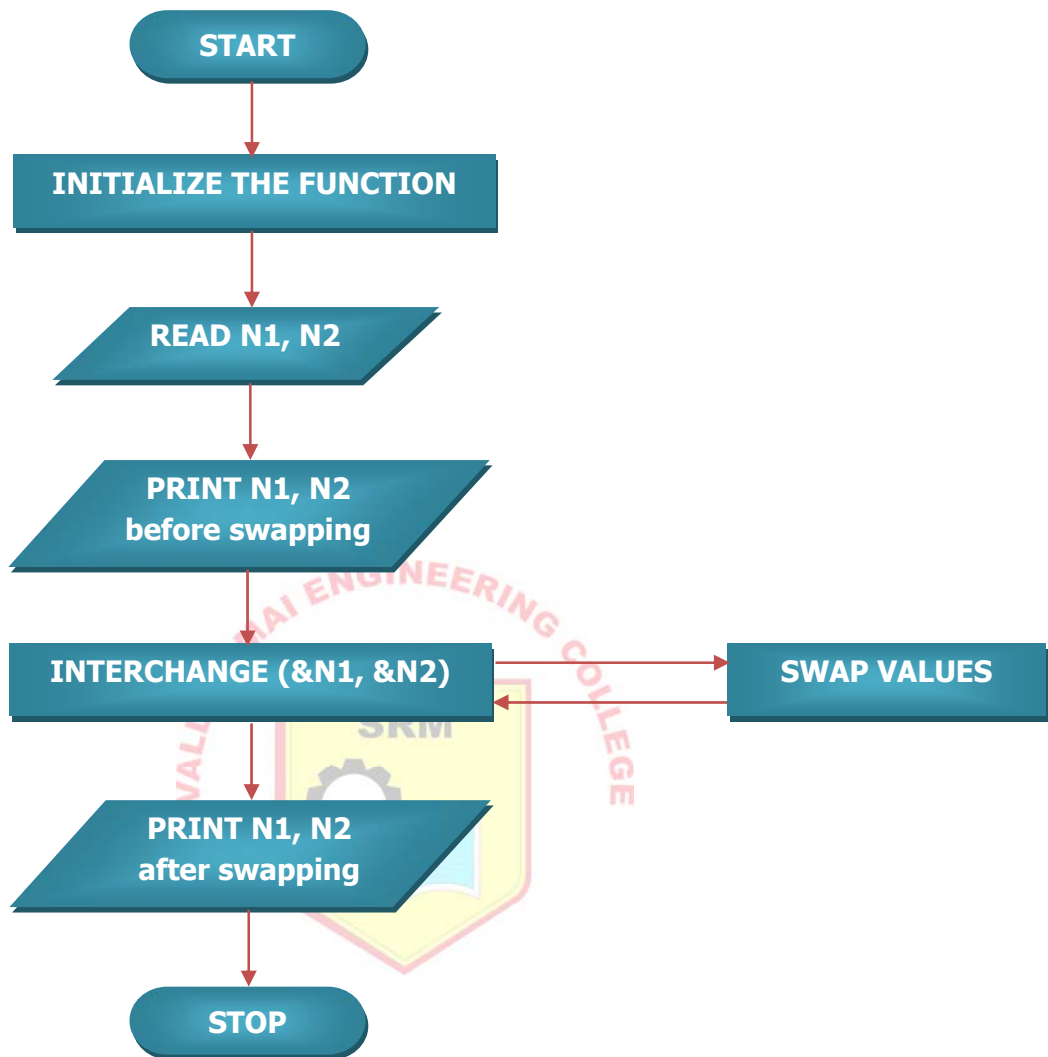
```

**RESULT:**

Thus a C program to swap two values with call by value was executed and the output was obtained.

L

FLOWCHART:



EX.NO : 8.2 SWAPPING OF TWO NUMBERS - CALL BY REFERENCE

DATE :

AIM:

To write a C Program to swap two values with call by reference.

ALGORITHM:

- Step 1: Start
- Step 2: Initialize the function INTERCHANGE(N1,N2) & declare the variables N1, N2
- Step 3: Get the two numbers N1, N2
- Step 4: Print N1 and N2 values before swapping
- Step 5: Exchange the values by calling the function INTERCHANGE(N1,N2) by passing the address (reference) of N1 & N2.
- Step 6: Print N1 and N2 values after swapping
- Step 7: Stop



L

OUTPUT:

```
D:\WINDOWS\system32\cmd.exe - tc

SWAPPING OF TWO NUMBERS - CALL BY REFERENCE
-----
Enter the first number N1: 50
Enter the second number N2: 70

The N1 and N2 values before swapping: 50 70
The N1 and N2 values after swapping: 70 50
```



PROGRAM:

```
#include<stdio.h>
void interchange(int *a, int *b);
void main()
{
    int n1=0,n2=0;
    clrscr();
    printf("\nSWAPPING OF TWO NUMBERS - CALL BY REFERENCE");
    printf("\n-----");
    printf("\n\nEnter the first number N1: ");
    scanf("%d",&n1);
    printf("\n\nEnter the second number N2: ");
    scanf("%d",&n2);
    printf("\n\nN1 and N2 values before swapping: %d %d\n",n1,n2);
    interchange(&n1,&n2);
    printf("\n\nN1 and N2 values after swapping: %d %d\n",n1,n2);
    getch();
}

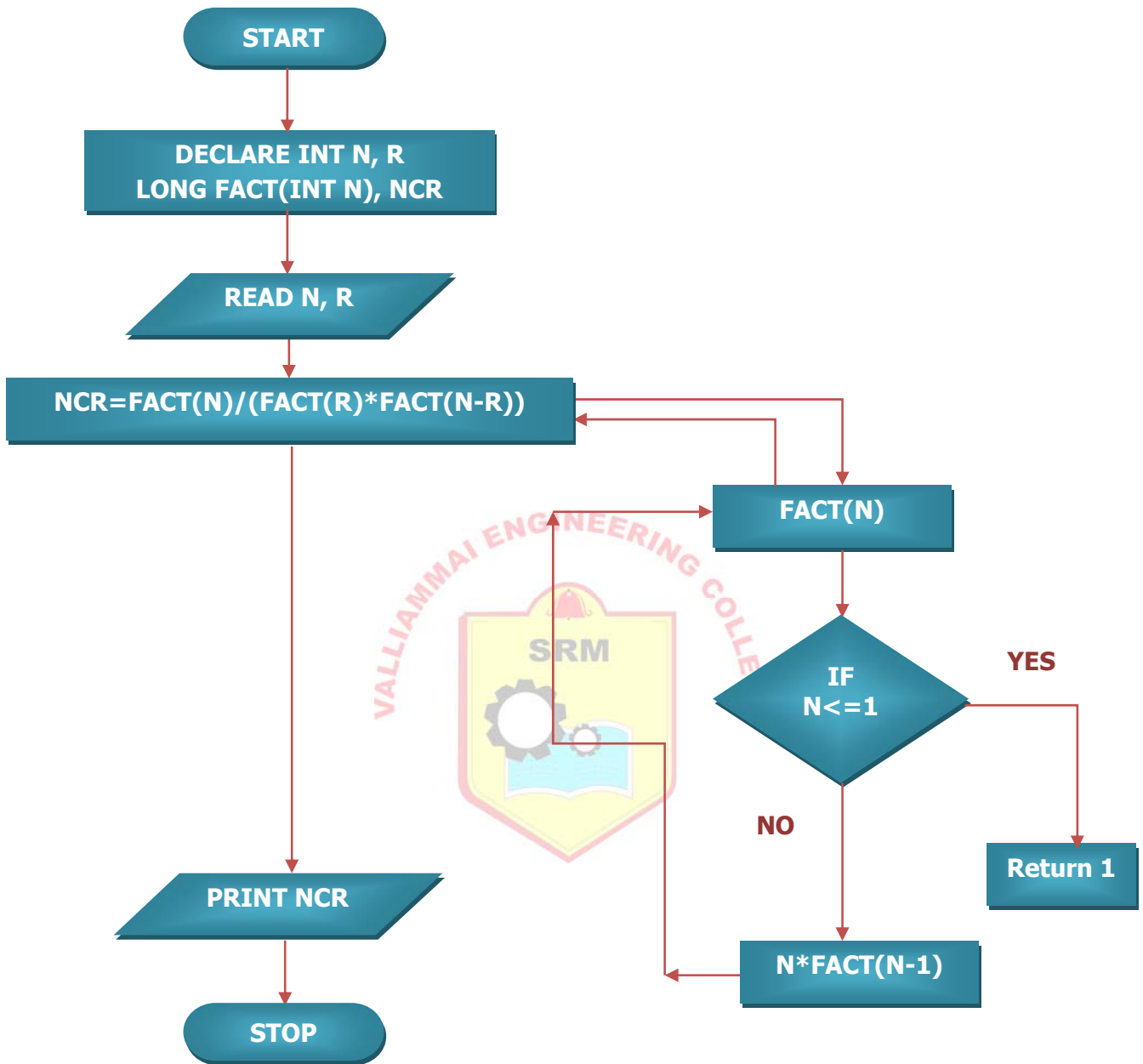
void interchange(int *a, int *b)
{
    int t;
    t=*a;
    *a=*b;
    *b=t;
}
```

RESULT:

Thus a C program to swap two values with call by reference was executed and the output was obtained.

L

FLOWCHART:



C PROGRAMMING EXERCISES**Programs using recursive functions****EX.NO: 9.1 COMBINATORICS (${}_n C_r$) USING RECURSION****DATE :****AIM:**

To write a C Program to find ${}_n C_r$ of a given numbers using recursive function.

$${}_n C_r = \frac{n!}{(r! * (n - r)!)}$$

ALGORITHM:

- Step 1: Start
- Step 2: Declare the variables LONG NCR, INT N, INT R and function LONG FACT (INT N).
- Step 3: Get the values of N and R.
- Step 4: Calculate $NCR = \text{FACT}(N) / (\text{FACT}(R) * \text{FACT}(N - R))$ by calling the user defined function LONG FACT (INT N).
- Step 5: Inside FACT()
 Step 5.1: If $N \leq 1$ then return 1, else
 Step 5.2: Multiply N with return value of FACT by passing value 1 as an argument.
 Step 5.3: Return the value of LONG FACT().
 Step 5.4: Print NCR by receiving the value of LONG FACT().
- Step 6: Stop.

L

OUTPUT:

```
D:\WINDOWS\system32\cmd.exe - tc

PRG TO FIND nCr USING RECURSION
-----
Enter the value of n: 5
Enter the value of r: 3

The value of 5C3 is 10.
```



PROGRAM:

```

#include<stdio.h>
void main()
{
    long fact(int n),ncr;
    int n,r;
    clrscr();
    printf("\nPRG TO FIND nCr USING RECURSION\n");
    printf("\n-----\n");
    printf("\nEnter the value of n: ");
    scanf("%d",&n);
    printf("\nEnter the value of r: ");
    scanf("%d",&r);
    ncr=fact(n)/(fact(r)*fact(n-r));
    printf("\nThe value of %dC%d is %ld",n,r,ncr);
    getch();
}

long fact(int n)
{
    if(n<=1)
        return (1);
    else
        return (n*fact(n-1));
}

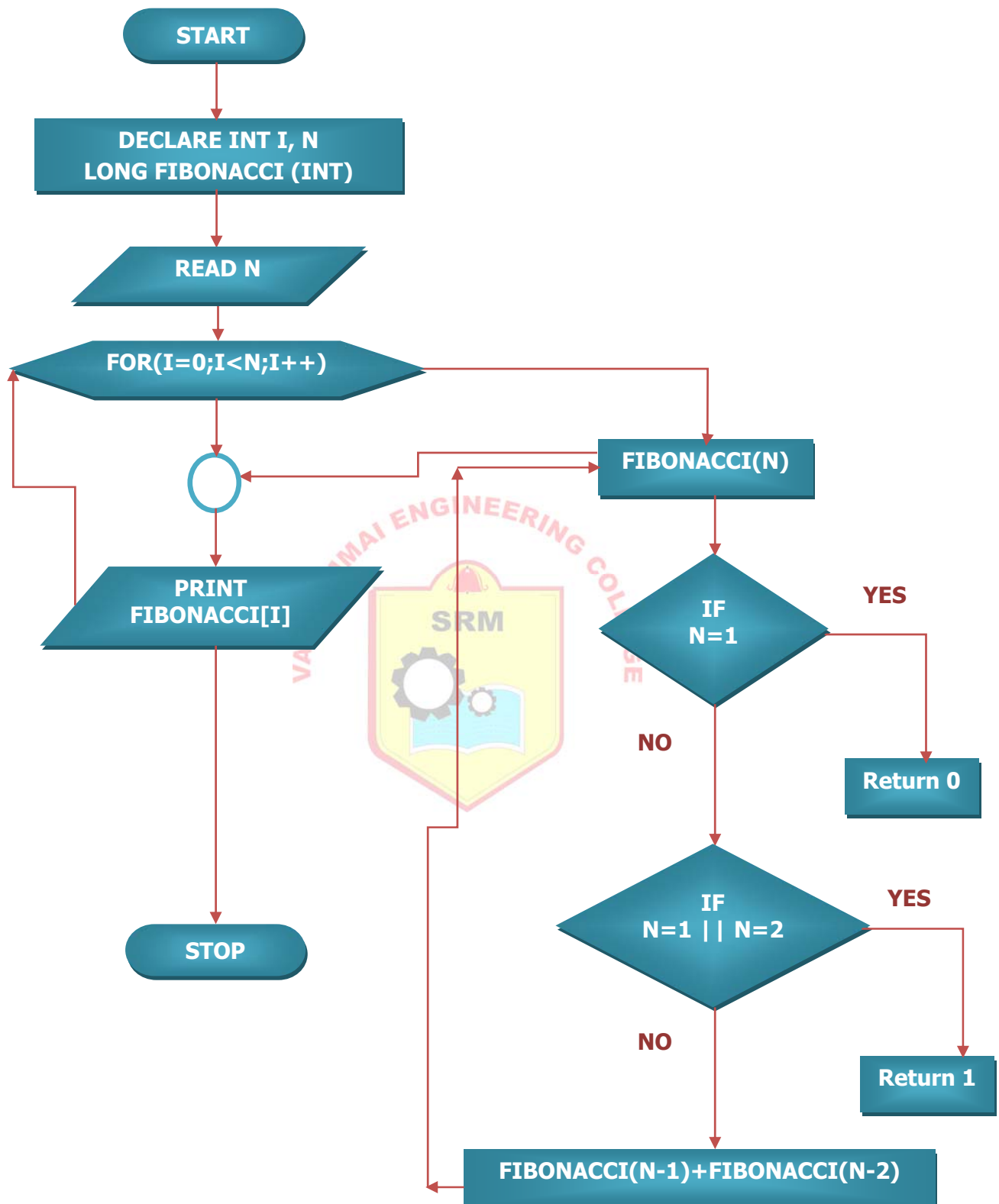
```

**RESULT:**

Thus a C program find the ${}_n C_r$ of a given numbers using recursion was executed and the output was obtained.



FLOWCHART:



EX.NO: 9.2 FIBONACCI SERIES UP TO A GIVEN TERM USING RECURSION**DATE :****AIM:**

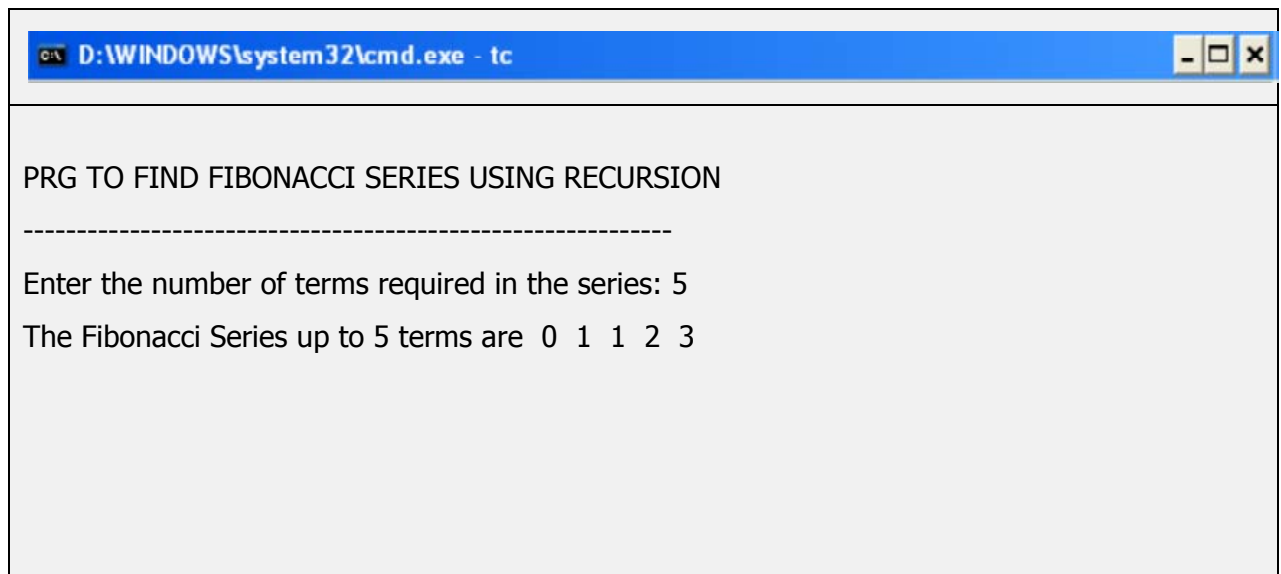
To write a C Program to generate a Fibonacci series using recursive function.

ALGORITHM:

- Step 1: Start
- Step 2: Declare the variables INT I, N and function LONG FIBONACCI (INT).
- Step 3: Get the values of N.
- Step 4: Set a for loop to calculate the Fibonacci series values up to N calling the user defined function FIBONACCI (I).
- Step 5: Inside FIBONACCI()
Step 5.1: If N=0 then return 0, else If N=1 or 2 then return 1, else
Step 5.2: Add FIBONACCI (N-1) value with return value of recursive function FIBONACCI (N-2).
Step 5.3: Return the value of LONG FIBONACCI().
Step 5.4: Print FIBONACCI by receiving the value of LONG FIBONACCI().
- Step 6: Stop.

L

OUTPUT:



```
D:\WINDOWS\system32\cmd.exe - tc

PRG TO FIND FIBONACCI SERIES USING RECURSION
-----
Enter the number of terms required in the series: 5
The Fibonacci Series up to 5 terms are 0 1 1 2 3
```



PROGRAM:

```

#include<stdio.h>
#include<conio.h>
long fibonacci(int);
void main()
{
    int i,n;
    clrscr();
    printf("\nPRG TO FIND FIBONACCI SERIES USING RECURSION\n");
    printf("\n-----\n");
    printf("\nEnter the number of terms required in the series: ");
    scanf("%d",&n);
    printf("\n\nThe Fibonacci Series up to %d terms are ",n);
    for(i=0;i<n;i++)
    {
        printf("%ld ",fibonacci(i));
    }
    getch();
}

long fibonacci(int n)
{
    if (n==0)
        return 0;
    else if(n==1 || n==2)
        return 1;
    else
        return (fibonacci(n-1)+fibonacci(n-2));
}

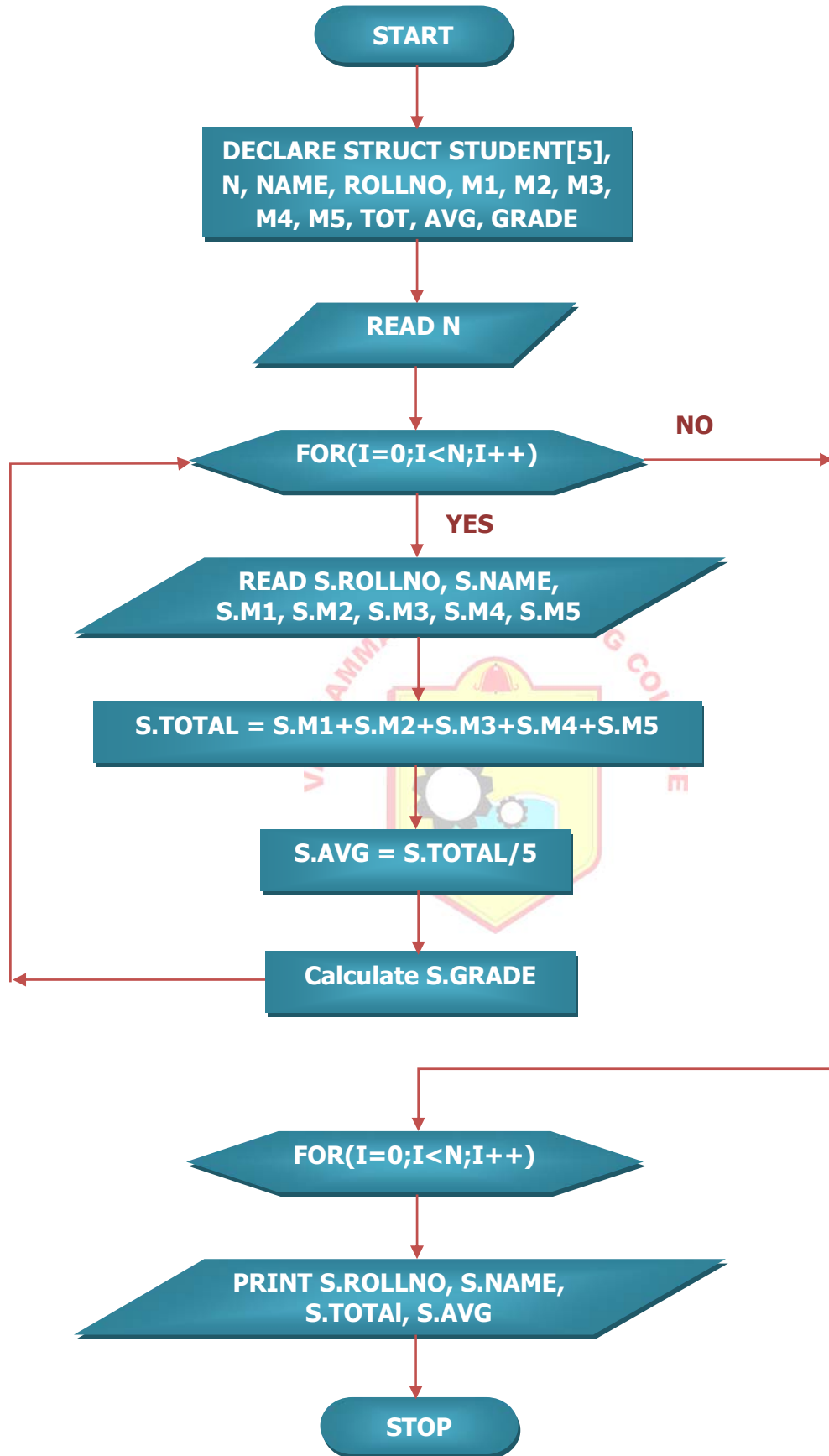
```

**RESULT:**

Thus a C program find the Fibonacci series up to a given terms using recursion was executed and the output was obtained.



FLOWCHART:



C PROGRAMMING EXERCISES**Programs using Structures and Unions****EX.NO: 10.1****STUDENTS RECORDS USING STRUCTURE****DATE :****AIM:**

To write a C Program to maintain students records using structure.

ALGORITHM:

- Step 1: Start
- Step 2: Declare the Structure Student with data members such as no, name, m1, m2, m3, m4, m5, total, avg and grade.
- Step 3: Get the number of students N
- Step 4: Set a for loop up to N to get the student details as input.
- Step 4.1: Enter the Student details such as Roll no, Name, Mark1, Mark2, Mark3, Mark4, Mark5.
- Step 4.2: Calculate Total and Avg, Grade.
- Step 4.3 Repeat Step 4.1 and 4.2 up to N.
- Step 5: Print Student Roll No, Name, Total, Avg and Grade.
- Step 6: Stop.

L



PROGRAM:

```
#include<stdio.h>
#include<conio.h>
struct student
{
    int no;
    char name[10],grade;
    int m1,m2,m3,m4,m5;
    float total,avg;
};
struct student s[5];
void main()
{
    int n,i;
    clrscr();
    printf("\nSTUDENTS RECORDS USING STRUCTURE");
    printf("\n\nEnter the no. of students:");
    scanf("%d",&n);
    printf("\nINSERT FIRST RECORD....\n");
    for(i=0;i<n;i++)
    {
        printf("\nRoll No.=");
        scanf("%d",&s[i].no);
        printf("Name=");
        scanf("%s",s[i].name);
        printf("Mark1=");
        scanf("%d",&s[i].m1);
        printf("Mark2=");
        scanf("%d",&s[i].m2);
        printf("Mark3=");
        scanf("%d",&s[i].m3);
        printf("Mark4=");
        scanf("%d",&s[i].m4);
        printf("Mark5=");
        scanf("%d",&s[i].m5);
```

L

OUTPUT:

```

D:\WINDOWS\system32\cmd.exe - tc

STUDENTS RECORDS USING STRUCTURE
-----
Enter the no. of students: 2

INSERT FIRST RECORD....
Roll No. = 101
Name    = Raj
Mark1   = 100
Mark2   = 80
Mark3   = 90
Mark4   = 90
Mark5   = 90

INSERT NEXT RECORD....
Roll No. = 102
Name    = Ram
Mark1   = 90
Mark2   = 70
Mark3   = 80
Mark4   = 90
Mark5   = 90

RECORD INSERTED SUCCESSFULLY....

STUDENT RECORD      NO.      NAME      TOTAL      AVG      GRADE
-----
STUDENT RECORD[0]   101      Raj       450        90       A
STUDENT RECORD[1]   102      Ram       420        84       A

```



```
s[i].total=s[i].m1+s[i].m2+s[i].m3+s[i].m4+s[i].m5;
s[i].avg=s[i].total/5;

if(s[i].avg<50)
    s[i].grade='D';
else if(s[i].avg<60)
    s[i].grade='C';
else if(s[i].avg<80)
    s[i].grade='B';
else s[i].grade='A';

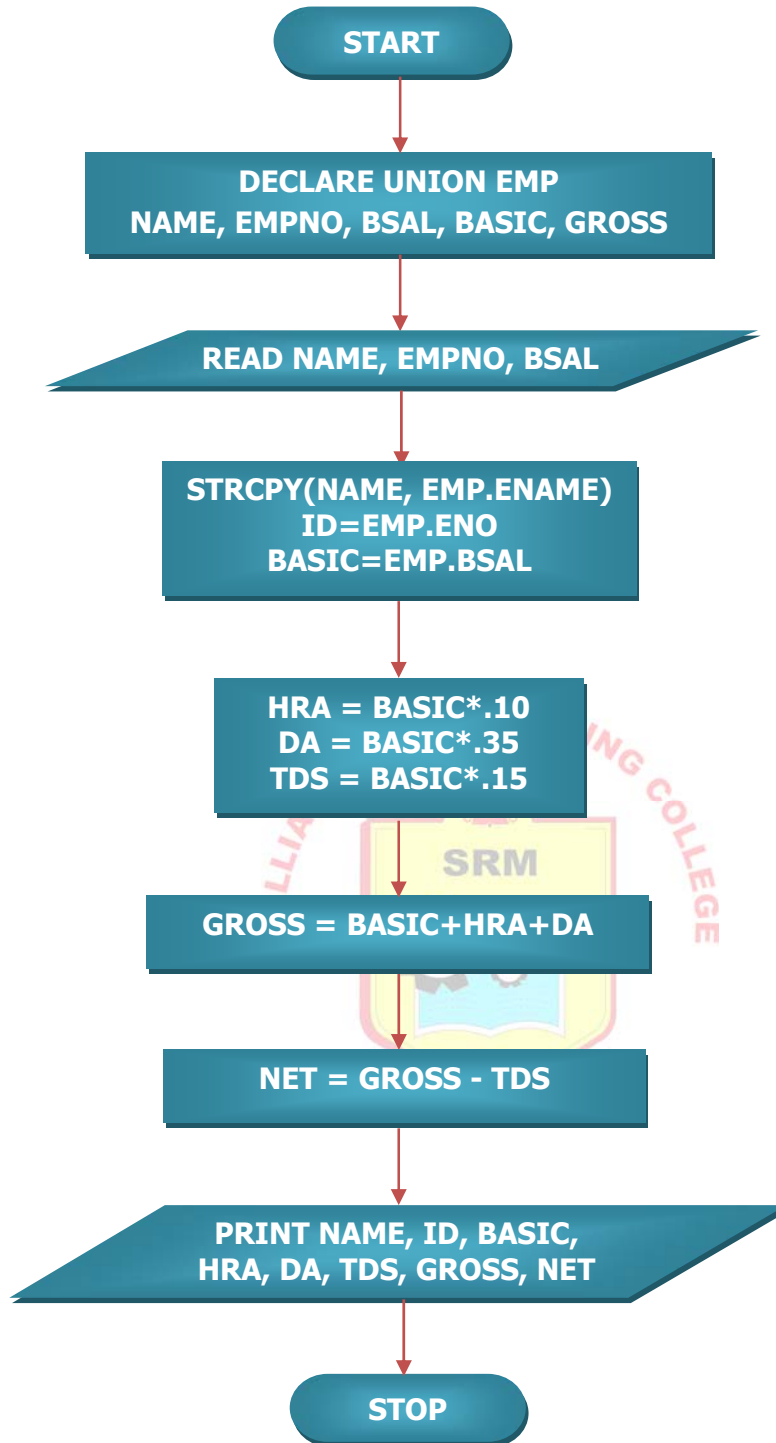
if(i!=n)
{
    printf("\n\nINSERT NEXT RECORD....\n");
}
else
{
    printf("\nRECORD INSERTED SUCCESSFULLY.....\n\n");
}
}
printf("\n");
printf("\nSTUDENT RECORD\t\tNO.\t\tNAME\t\tTOTAL\t\tAVG\t\tGRADE\n");
printf("\n-----\t\t---\t\t---\t\t----\t\t---\t\t-----\n");
for(i=0;i<n;i++)
{
    printf("\nSTUDENT RECORD[%d]:\t",i);
    printf("%d\t%s\t%.2f\t%.2f\t %c",s[i].no,s[i].name,s[i].total,s[i].avg,s[i].grade);
}
getch();
}
```

RESULT:

Thus a C program to implement the student records using structure was executed and the output was obtained.



FLOWCHART:



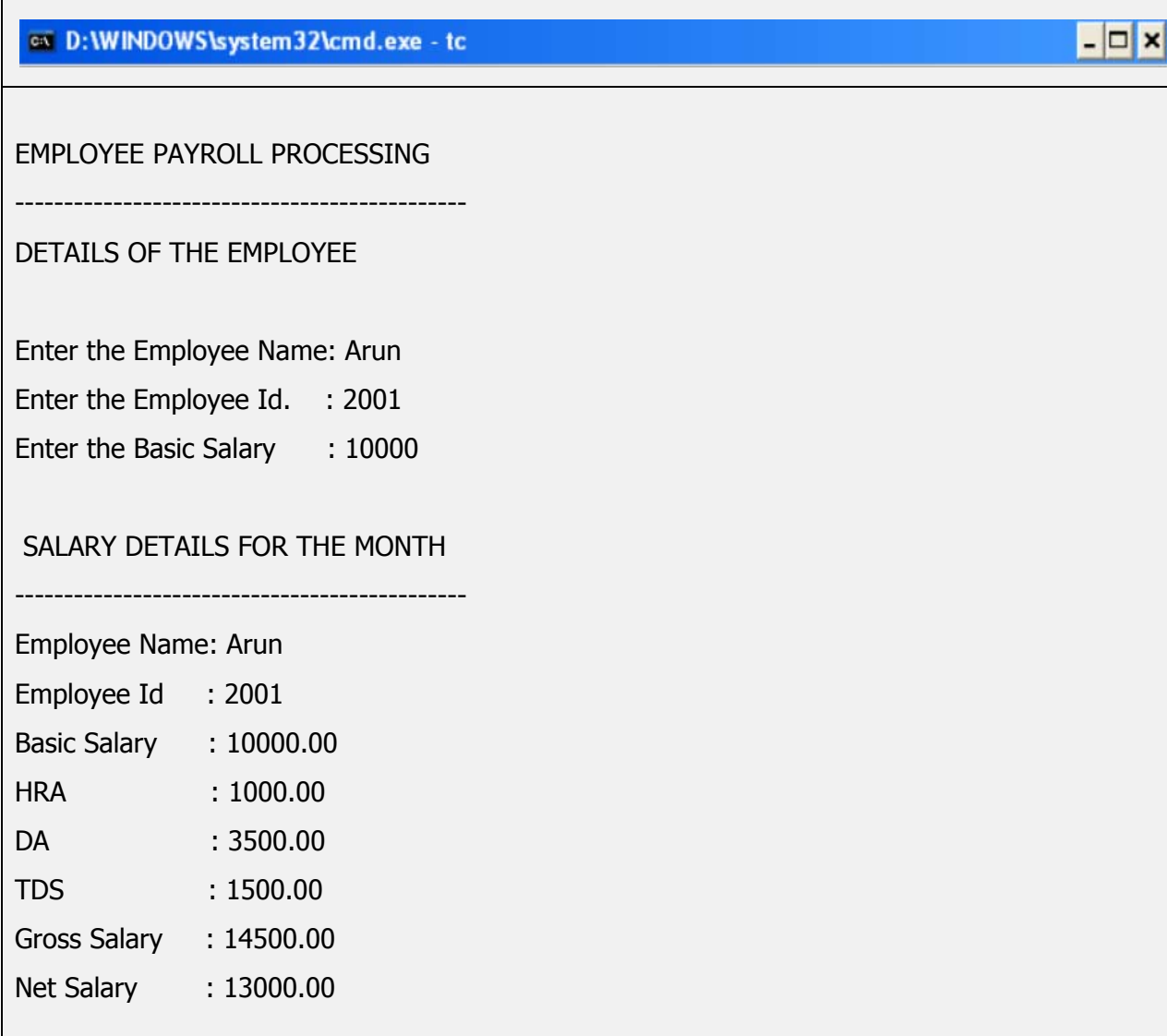
EX.NO: 10.2**PAYROLL PROCESSING USING UNION****DATE :****AIM:**

To write a C Program to create payroll processing using union.

ALGORITHM:

- Step 1: Start
- Step 2: Declare the Union Employee with data members such as name, eno, basic salary, net salary, gross.
- Step 3: Get the details of an employee.
- Step 4: Enter the employee details such as Name, Emp No and Basic salary.
- Step 5: Copy the emp.ename to Name by using `STRCPY(NAME,EMP.ENAME)` and assign `ID=EMP.ENO, BASIC=EMP.BSAL`.
- Step 6: Calculate $HRA=10\%*BASIC$, $DA=35\%*BASIC$, and $TDS=15\%*BASIC$
- Step 7: Calculate $GROSS=BASIC+HRA+DA$ and $NET\ SALARY=GROSS-TDS$
- Step 8: Print Employee salary details such as Name, Id, Basic, HRA, DA, TDS, GROSS and NET SALARY.
- Step 9: Stop.

L

OUTPUT:

```
D:\WINDOWS\system32\cmd.exe - tc

EMPLOYEE PAYROLL PROCESSING
-----
DETAILS OF THE EMPLOYEE

Enter the Employee Name: Arun
Enter the Employee Id.   : 2001
Enter the Basic Salary   : 10000

SALARY DETAILS FOR THE MONTH
-----

Employee Name: Arun
Employee Id   : 2001
Basic Salary  : 10000.00
HRA          : 1000.00
DA           : 3500.00
TDS          : 1500.00
Gross Salary : 14500.00
Net Salary   : 13000.00
```

PROGRAM:

```

#include<stdio.h>
#include<conio.h>
union employee
{
    char ename[30];
    int eno;
    float bsal;
};
void main()
{
    union employee emp;
    char name[30];
    int id;
    float basic,hra,da,tds,net,gross;
    clrscr();
    printf("\nEMPLOYEE PAYROLL PROCESSING\n");
    printf("\n-----\n");
    printf("\nDETAILS OF THE EMPLOYEE\n\n");
    printf("\nEnter the Employee Name: ");
    scanf("%s",emp.ename);
    strcpy(name,emp.ename);
    printf("\nEnter the Employee Id: ");
    scanf("%d",&emp.eno);
    id=emp.eno;
    printf("\nEnter the Basic Salary: ");
    scanf("%f",&emp.bsal);
    basic=emp.bsal;
    hra=basic*.10;
    da=basic*.35;
    tds=basic*.15;
    gross=basic+hra+da;
    net=gross-tds;
    printf("\n\nSALARY DETAILS FOR THE MONTH\n");
    printf("\n-----\n");
    printf("\n\nEmployee Name\t: %s",name);
    printf("\n\nEmployee No.\t: %d",id);
    printf("\n\nBasic salary\t: %.2f",basic);
    printf("\n\nHRA\t\t: %.2f",hra);
    printf("\n\nDA\t\t: %.2f",da);
    printf("\n\nTDS\t\t: %.2f",tds);
    printf("\n\nGross Salary\t: %.2f",gross);
    printf("\n\nNet Salary\t: %.2f",net);
    getch();
}

```

RESULT:

Thus a C program to implement the payroll processing using union was executed and the output was obtained.



Additional C Programs for exercise

1. Program to print ODD numbers from 1 to 100.

```
#include<stdio.h>
#include<conio.h>
void main( )
{
int i;
clrscr( );
for (i=1; i<=100; i+=2)
{
printf("%d\n",i);
}
getch( );
}
```

2. Program to print natural numbers from 1 to 50 in reverse order.

```
#include<stdio.h>
#include<conio.h>
void main( )
{
int i;
clrscr( );
for (i=50; i>=1; i--)
{
printf("%d\n",i);
}
getch( );
}
```



3. Program to print sum of the natural numbers from 1 to n.

```
#include<stdio.h>
#include<conio.h>
void main( )
{
int n,sum=0,i;
clrscr( );
printf("Enter an integer: ");
scanf("%d",&n);
for (i=1; i<=n; i++)
{
sum=sum+i;
}
printf("The Sum of natural numbers from 1 to n is %d\n",sum);
getch( );
}
```

4. Program to read two numbers and print the sum of given two numbers.

```
#include<stdio.h>
#include<conio.h>
void main()
{
int a,b, sum;
clrscr ();
printf ("Enter value for A ; ");
scanf ("%d",&a);
printf("Enter value for B ;");
scanf("%d",&b);
sum=a+b;
printf("Sum Of Given Two Numbers are %d", sum);
getch();
}
```

5. Program to accept student roll no, marks in 3 subjects and calculate total, average.

```
#include<stdio.h>
#include<conio.h>
void main()
{
int r,b,c,d, tot, avg;
clrscr();
printf ("ENTER STUDENT RNO : ");
scanf ("%d",&r);
printf("ENTER FIRST SUBJECT MARKS :");
scanf("%d",&b);
printf("ENTER SECOND SUBJECT MARKS :");
scanf("%d",&c);
printf("ENTER THIRD SUBJECT MARKS :");
scanf("%d",&d);
tot=b+c+d;
avg=tot/3;
printf("\n\n\t\t SRMVEC - CHENNAI\n\n");
printf("\t STUDENT R.NO : %d ",r);
printf("\t FIRST SUBJECT MARKS :%d ",b);
printf("\t SECOND SUBJECT MARKS ;%d ",C);
printf("\t THIRD SUBJECT MARKS :%d ",d);
printf("\t AVERAGE MARKS : %d", avg);
getch();
}
```



6. Program to accept a year and check whether the given year is leap year or not.

```
#include<stdio.h>
#include<conio.h>
void main( )
{
int y;
clrscr( );
printf("Enter a year:");
scanf("%d",&y);
if(y%4==0 && y%100!=0 || y%400==0)
    printf("The above given year is a leap year");
else
    printf("The above given year is not a leap year");
getch();
}
```

7. Program to accept a number and print mathematical table of the given number.

```
#include<stdio.h>
#include<conio.h>
void main( )
{
int i,t;
clrscr( );
printf("Table to Print:");
scanf("%d",&t);
for (i=1; i<=20; i++)
{
    printf("\n%d*%d=%d",t,i,i*t);
}
getch( );
}
```

**8. Program to print numeric pyramid.**

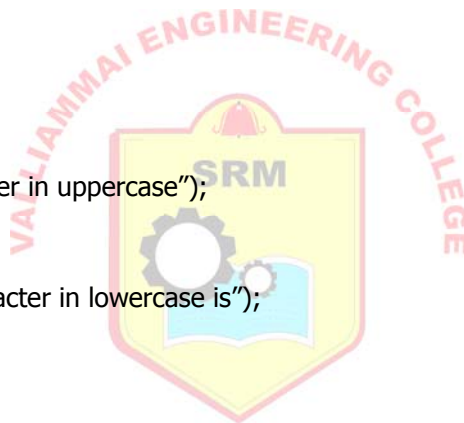
```
#include<stdio.h>
#include<conio.h>
void main()
{
int i,j;
clrscr( );
for(i=1;i<=5;i++)
{
    for(j=1;j<=i;j++)
    {
        printf("%d",j);
        printf("\n");
    }
}
getch();
}
```

9. Program to find biggest of two no by using ternary operator.

```
#include<stdio.h>
#include<conio.h>
void main( )
{
int a,b,big;
clrscr( );
printf("Enter the value of a: ");
scanf("%d",&a);
printf("Enter the value of b");
scanf("%d",&b);
big=(a>b)?a:b;
printf("Biggest of the given numbers is %d",big);
getch();
}
```

10. Program to accept a character in the uppercase and print in lower case.


```
#include<stdio.h>
#include<conio.h>
void main( )
{
char ch,c1;
clrscr( );
printf("Enter a character in uppercase");
ch=getchar();
c1=ch+32;
printf("The given character in lowercase is");
putchar(c1);
getch();
}
```

**11. Program to accept a character in any case and print in another case.**

```
#include<stdio.h>
#include<conio.h>
void main( )
{
char ch,c1;
clrscr( );
printf("Enter a char in anycase");
ch=getchar();
if(ch>=65 && ch<=90)
c1=ch+32;
else
if(ch>=97 && ch<=122)
c1=ch-32;
printf("The given char in another case is");
putchar(c1);
getch();
}
```

12. Program to convert Celsius to Fahrenheit

```
#include<stdio.h>
#include<conio.h>
void main()
{
    float cel,faren;
    clrscr();
    printf("\nCONVERSION OF CELSIUS INTO FAHRENHEIT");
    printf("\n-----");
    printf("\n\nEnter the celsius value:");
    scanf("%f",&cel);
    faren=(1.8*cel)+32;
    printf("\nThe equivalent fahrenheit value of the given %5.2f celsius value is
%5.2f",cel,faren);
    getch();
}
```


13. Program to largest and smallest number in a given array.


```
#include<stdio.h>
#include<conio.h>
void main()
{
    int a[10],large,small,n,i;
    clrscr();
    printf("\nLARGEST AND SMALLEST NUMBER IN A GIVEN ARRAY");
    printf("\n-----");
    printf("\n\nEnter the no of elements");
    scanf("%d",&n);
    printf("\n\nEnter the elements");
    for(i=0;i<n;i++)
    scanf("%d",&a[i]);
    large=small=a[0];
    for(i=1;i<n;i++)
    {
        if(a[i]>large)
        large=a[i];
        if(a[i]<small)
        small=a[i];
    }

    printf("\nThe largest no is %d and the smallest no is %d",large,small);
    getch();
}
```

14. Program to accept a string in any case and print it by another case.

```
#include<stdio.h>
#include<conio.h>
void main( )
{
char ch;
clrscr( );
printf("Enter a string :");
while(( ch=getchar( ))!='\n')
{
if(ch>='A' && ch<='Z')
putchar(ch+32);
else
if(ch>='a' && ch<='z')
putchar(ch-32);
else
putchar(ch);
}
getch( );
}
```

15. Program to accept a string and print the reverse of the given string by using for loop.


```
#include<stdio.h>
#include<conio.h>
void main( )
{
int i,j;
char name[80];
clrscr( );
printf(" Enter a string");
gets(name);
for(i=0;i<80 && ((name [i]= getchar())!='\n');i++); if(name[i]= '\n')
name[i]='\0';
for(j=i;j>=0;j--)
putchar(name[j]);
printf("is the reverse of given string");
getch( );
}
```

16. Program to read date, month, year and print the next day's date, month, year.

```
#include<stdio.h>
#include<conio.h>
void main( )
{
int month[12]={31,28,31,30,31,30,31,31,30,31,30,31}; int d,m,y,nd,nm,ny,ndays;
clrscr( );
printf("Enter the date, month, year");
scanf("%d%d%d",&d,&m,&y);
ndays=month[m-1];
if(m==2)
```

```

{
if(y%100==0)
{
if(y%400==0)
ndays=29;
}
else
if(y%4==0)
ndays=29;
}
nd=nd+1;
nm=m;
ny=y;
if(nd>ndays)
{
nd=1;
nm++;
}
if(nm>12)
{
nm=1;
ny++;
}
printf("Given date is %d:%d:%d\n",d,m,y);
printf("The next day's date is %d:%d:%d",nd,nm,ny);
getch( );
}

```

17. Program to interchange two values using pointers.

```

#include<stdio.h>
#include<conio.h>
void interchange(int *x,int *y);
void main( )
{
int a,b;
clrscr( );
printf("enter values of a and b");
scanf("%d%d",&a,&b);
interchange(&a,&b);
}
void interchange(x,y)
int *x,*y;
{
int t;
t=*x;
*x=*y;
*y=t;
printf("%d=x, %d=y",*x,*y);
getch( );
}

```

18. Program to print "PASCAL TRIANGLE".

```

#include<stdio.h>
#include<conio.h>
void main()
{
int n,p=1,q,num,sp;
clrscr( );
printf("Enter the number of rows");
scanf("%d",&n);
for(p=0;p<=n;p++)
{
for(sp=1;sp<=40-(3*p);sp++)
printf(" ");
for(q=0;q<n;q++)
{
if((q==q)|| (q==0))
num=1;
else
num=num*((q-q)+1)/q;
printf("%2d",num);
printf("\n");
}
}
getch( );
}

```

**19. Program to check whether a given number is perfect or not.**

```

#include<stdio.h>
#include<conio.h>
void main( )
{
int i,n,s=0;
clrscr();

printf("enter the number");
scanf("%d",&n);
for(i=1;i<n/2;i++)
if(n%i==0)
s+=i;
if(s==n)
printf("The number is perfect number.");
else
printf("The number is not perfect number. ");
getch( );
}

```

20. Program to print Armstrong number between 1 to 500.

```
#include<stdio.h>
#include<conio.h>
void main( )
{
int i,n,s,r;
clrscr( );
for(i=1;i<=500;i++)
{
n=i;
s=0;
while(n>0)
{
r=n%10;
s=s+(r*r*r);
n=n/10;
}
if(i==s)
printf("\n%d",s);
}
getch();
}
```

21. Program to print the floyd's triangle.

```
#include<stdio.h>
#include<conio.h>
void main( )
{
int i,n,s,r k=1;
clrscr( );
printf("enter a number of rows");
scanf("%d",&n);
for(i=1;i<=n;i++)
{
for(s=1;s<=40-i;s++)
printf(" ");
for(j=1;j<=i;j++)
printf("%3d",k++);
printf("\n");
}
getch( );
}
```



22. Program to add two matrices.

```

#include<stdio.h>
#include<conio.h>
void main()
{
int a[25][25],b[25][25],c[25][25],i,j,m,n;
clrscr();

printf("\nMATRIX ADDITION");
printf("\n~~~~~ ~~~~~");

printf("\nEnter the row n column of 2 matrices:");
scanf("%d%d",&m,&n);

printf("\nEnter the Elements of A matrix");
printf("\n");
for(i=0;i<m;i++)
{
for(j=0;j<n;j++)
scanf("%d",&a[i][j]);
}

printf("\nEnter the Elements of B matrix");
printf("\n");
for(i=0;i<m;i++)
{
for(j=0;j<n;j++)
scanf("%d",&b[i][j]);
}

printf("\nThe elements of A matrix are:");
for(i=0;i<m;i++)
{
printf("\n");
for(j=0;j<n;j++)
printf("\t%d",a[i][j]);
}

printf("\nThe elements of B matrix are:");
for(i=0;i<m;i++)
{
printf("\n");
for(j=0;j<n;j++)
printf("\t%d",b[i][j]);
}

printf("\nThe resultant matrix:");
for(i=0;i<m;i++)
{
printf("\n");
for(j=0;j<n;j++)
{
c[i][j]=a[i][j]+b[i][j];
printf("\t %d",c[i][j]);
}
}
getch();
}

```



23. Program to subtract a matrix from another matrix.

```

#include<stdio.h>
#include<conio.h>
void main()
{
    int a[25][25],b[25][25],c[25][25],i,j,m,n;
    clrscr();

    printf("\nMATRIX SUBTRACTION");
    printf("\n~~~~~ ~~~~~~");

    printf("\nEnter the row n column of 2 matrices:");
    scanf("%d%d",&m,&n);

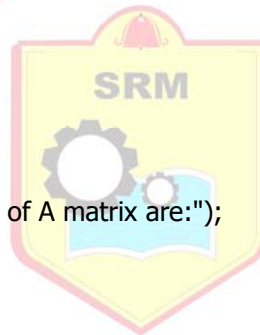
    printf("\nEnter the Elements of A matrix");
    printf("\n");
    for(i=0;i<m;i++)
    {
        for(j=0;j<n;j++)
            scanf("%d",&a[i][j]);
    }

    printf("\nEnter the Elements of B matrix");
    printf("\n");
    for(i=0;i<m;i++)
    {
        for(j=0;j<n;j++)
            scanf("%d",&b[i][j]);
    }

    printf("\nThe elements of A matrix are:");
    for(i=0;i<m;i++)
    {
        printf("\n");
        for(j=0;j<n;j++)
            printf("\t%d",a[i][j]);
    }

    printf("\nThe elements of B matrix are:");
    for(i=0;i<m;i++)
    {
        printf("\n");
        for(j=0;j<n;j++)
            printf("\t%d",b[i][j]);
    }
    printf("\nThe resultant matrix:");
    for(i=0;i<m;i++)
    {
        printf("\n");
        for(j=0;j<n;j++)
        {
            c[i][j]=a[i][j]-b[i][j];
            printf("\t %d",c[i][j]);
        }
    }
    getch();
}

```



24. Program to transpose a matrix.

```

#include<stdio.h>
#include<conio.h>
void main()
{
    int a[25][25],b[25][25],i,j;
    clrscr();
    printf("\nMATRIX TRANSPOSE");
    printf("\n~~~~~ ~~~~~");
    printf("\nEnter the row n column of 2 matrices:");
    scanf("%d%d",&m,&n);
    printf("\nEnter the Elements of A matrix");
    printf("\n");
    for(i=0;i<m;i++)
    {
        for(j=0;j<n;j++)
            scanf("%d",&a[i][j]);
    }

    printf("\nThe elements of A matrix are:");
    for(i=0;i<m;i++)
    {
        printf("\n");
        for(j=0;j<n;j++)
            printf("\t%d",a[i][j]);
    }
    printf("\nThe resultant matrix:");
    for(i=0;i<m;i++)
    {
        printf("\n");
        for(j=0;j<n;j++)
        {
            c[i][j]=a[j][i];
            printf("\t %d",c[i][j]);
        }
    }
    getch();
}

```

25. Program to find the row sum and column sum of a given matrix.

```

#include<stdio.h>
#include<conio.h>
void main()
{
int mat[10][10];
int i,j;
int m,n;
int sumrow,sumcol;

clrscr();

printf("\nTO FIND THE ROW SUM AND COLUMN SUM OF A GIVEN MATRIX:");
printf("\n-- -----");
printf("\nEnter the order of matrix:");
scanf("%d%d",&m,&n);
printf("\nEnter elements of a matrix:");
for(i=0;i<m;i++)
{
for(j=0;j<n;j++)
scanf("%d",&mat[i][j]);
}
printf("\n\nOUTPUT:");
printf("\n-----");
for(i=0;i<m;i++)
{
sumrow=0;
for(j=0;j<n;j++)
sumrow=sumrow+mat[i][j];
printf("\nTHE SUM OF %d ROW IS %d",i+1,sumrow);
}
printf("\n-----");
for(j=0;j<n;j++)
{
sumcol=0;
for(i=0;i<m;i++)
sumcol=sumcol+mat[i][j];
printf("\nTHE SUM OF %d COLUMN IS %d",j+1,sumcol);
}
printf("\n-----");
getch();
}

```



26. Program to find whether a number is divisible by '11' or not without actual division.

```

#include<stdio.h>
#include<conio.h>
#include<math.h>
void main( )
{
int a,b,n,evensum=0,oddsum=0,div;
clrscr( );
printf("Enter a number");
scanf("%d",&n);
a=n;
b=n/10;
while(a>0)
{
oddsum=oddsum+(a%10);
a=a/10;
}
while(b>0)
{
evensum=evensum+(b%10);
b=b/10;
}
div=abs(evensum-oddsum);
if(div%11==0)
    printf("The number is divisible by 11");
else
    printf("The number is not divisible by 11");
getch();
}

```

27. Program to read a string and print the first two characters of each word in the string.

```

#include<stdio.h>
#include<conio.h>
void main( )
{
char s[100];
int i,l;
clrscr( );
printf("Enter a string");
gets(s);
l=strlen(s);
for(i=0;i<l;i++)
{
if(s[i]!=' ' && s[i]!='\n')
{
printf("%c %c",s[i],s[i+1])
i=i+2;
while(s[i]!=' ')
i++;
}
}
getch( );
}

```

28. Program to accept a string in lower case and print first character of each word in upper case.

```
#include<conio.h>
#include<stdio.h>
void main( )
{
char str1[80];
int length,i;
clrscr( );
printf("enter a string; ");
length=getline(str1,80);
for(i=0;i<length;i++)
{
str1[0]-=32;
if(str1[i]==')
str1[i+1]-=32;
printf("%c".str1[i]);
}
getch();
}
int getline(char line [], int lim)
{
int i;
for(i=0;i<lim && ((line[i]=getchar( ))!=\n');i++); if(line[i]= \n')
line[i]=\0';
return i;
}
```

29. Program to check whether two strings are anagrams or not.

```
#include <stdio.h>
int check_anagram(char [], char []);
int main()
{
char a[100], b[100];
int flag;
printf("Enter first string\n");
gets(a);
printf("Enter second string\n");
gets(b);
flag = check_anagram(a, b);
if (flag == 1)
printf("\n%s" and "%s" are anagrams.\n", a, b);
else
printf("\n%s" and "%s" are not anagrams.\n", a, b);
return 0;
}
int check_anagram(char a[], char b[])
{
int first[26] = {0}, second[26] = {0}, c = 0;
while (a[c] != '\0')
{
first[a[c]-'a']++;
c++;
}
c = 0;
}
```

L

```

while (b[c] != '\0')
{
    second[b[c]-'a']++;
    c++;
}
for (c = 0; c < 26; c++)
{
    if (first[c] != second[c])
        return 0;
}
return 1;
}


```

30. Program to find HCF and LCM.

```

#include <stdio.h>
int main() {
int a, b, x, y, t, gcd, lcm;
printf("Enter two integers\n");
scanf("%d%d", &x, &y);
a = x;
b = y;
while (b != 0)
{
    t = b;
    b = a % b;
    a = t;
}
gcd = a;
lcm = (x*y)/gcd;
printf("Greatest common divisor of %d and %d = %d\n", x, y, gcd);
printf("Least common multiple of %d and %d = %d\n", x, y, lcm);
return 0;
}

```


31. Program to convert decimal to binary.

```

#include <stdio.h>
void main()
{
int n, c, k;
printf("Enter an integer in decimal number system\n");
scanf("%d", &n);
printf("%d in binary number system is:\n", n);
for (c = 31; c >= 0; c--)
{
    k = n >> c;
    if (k & 1)
        printf("1");
    else
        printf("0");
}
printf("\n");
}

```

32. Program to generate 10 random numbers between 1 and 100.

```
#include <stdio.h>
#include <stdlib.h>
int main() {
    int c, n;
    printf("Ten random numbers in [1,100]\n");
    for (c = 1; c <= 10; c++)
    {
        n = rand()%100 + 1;
        printf("%d\n", n);
    }
    return 0;
}
```

33. Program to print current system date.

```
#include <stdio.h>
#include <conio.h>
#include <dos.h>
int main()
{
    struct date d;
    getdate(&d);
    printf("Current system date is %d/%d/%d",d.da_day,d.da_mon,d.da_year);
    getch();
    return 0;
}.
```

34. Program to calculate Standard Deviation.

```
#include <stdio.h>
#include <math.h>
float standard_deviation(float data[], int n);
int main()
{
    int n, i;
    float data[100];
    printf("Enter number of datas( should be less than 100): ");
    scanf("%d",&n);
    printf("Enter elements: ");
    for(i=0; i<n; ++i)
        scanf("%f",&data[i]);
    printf("\n");
    printf("Standard Deviation = %.2f", standard_deviation(data,n));
    return 0;
}
float standard_deviation(float data[], int n)
{
    float mean=0.0, sum_deviation=0.0;
    int i;
    for(i=0; i<n;++i)
    {
        mean+=data[i];
    }
}
```

L

```

    mean=mean/n;
    for(i=0; i<n;++i)
    sum_deviation+=(data[i]-mean)*(data[i]-mean);
    return sqrt(sum_deviation/n);
}

```

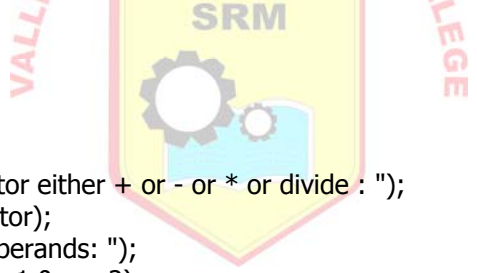
35. Program to convert Hexadecimal to Octal and Vice Versa.

```

#include <stdio.h>
int main()
{
    int n;
    printf("Enter an octal number: ");
    scanf("%o",&n);    /*Takes number in octal format. */
    /* %o and %x will display the number in octal format and hexadecimal form respectively. */
    printf("%o in octal = %x in hexadecimal", n, n);
    printf("\nEnter an hexadecimal number: ");
    scanf("%x",&n);    /* Takes number in hexadecimal format.*/
    printf("%x in hexadecimal = %o in octal", n, n);
    return 0;
}

```

36. Program to make a simple Calculator to Add, Subtract, Multiply or Divide.



```

# include <stdio.h>
int main()
{
    char operator;
    float num1,num2;
    printf("Enter operator either + or - or * or divide : ");
    scanf("%c",&operator);
    printf("Enter two operands: ");
    scanf("%f%f",&num1,&num2);
    switch(operator)
    {
    case '+':
        printf("num1+num2=%.2f",num1+num2);
        break;
    case '-':
        printf("num1-num2=%.2f",num1-num2);
        break;
    case '*':
        printf("num1*num2=%.2f",num1*num2);
        break;
    case '/':
        printf("num2/num1 = %.2f",num1/num2);
        break;
    default:
        /* If operator is other than +, -, * or /, error message is shown */
        printf("Error! operator is not correct");
        break;
    }
    return 0;
}

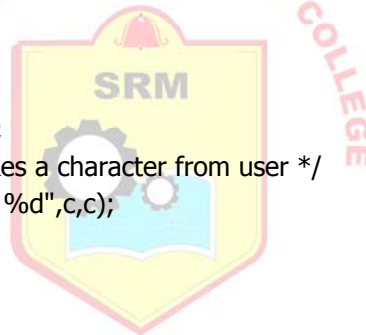
```


37. Program to calculate the Power of a Number using Recursion.

```
#include <stdio.h>
int power(int n1,int n2);
int main()
{
    int base, exp;
    printf("Enter base number: ");
    scanf("%d",&base);
    printf("Enter power number(positive integer): ");
    scanf("%d",&exp);
    printf("%d^%d = %d", base, exp, power(base, exp));
    return 0;
}
int power(int base,int exp)
{
    if ( exp!=1 )
        return (base*power(base,exp-1));
}
```

38. Program to find the ASCII value of a Character.

```
#include <stdio.h>
int main(){
    char c;
    printf("Enter a character: ");
    scanf("%c",&c); /* Takes a character from user */
    printf("ASCII value of %c = %d",c,c);
    return 0;
}
```

**39. Program to find biggest of four no by using ternary numbers.**

```
#include<stdio.h>
#include<conio.h>
void main( )
{
    int a,b,c,d,big;
    clrscr( );
    printf("\nenter value a");
    scanf("%d",&a);
    printf("\nenter the value of b");
    scanf("%d",&b);
    printf("\nenter the value of c");
    scanf("%d",&c);
    printf("\nenter the value of d");
    scanf("%d",&d);
    big=(a>b)?(a>c)?(a>d)?a:d:(c>d)?c:d:(b>c)?(b>d)?b:d:(c>d)?c:d;
    printf("Biggest of the given 4 numbers is %d",big);
    getch();
}
```

40. Program to calculate sum of the individual digits of a number.

```
#include<stdio.h>
#include<conio.h>
void main()
{
int n,sum,r;
clrscr();
printf("Enter an Integer: ");
scanf("%d",&n);
sum=0;
while(n>0)
{
r=n%10;
sum=sum+r;
n=n/10;
}
printf("Sum of Digits: %d",sum);
getch();
}
```



C Language Questions and Answers

1. What is C language?

C is a programming language developed at AT&T's Bell Laboratories of USA in 1972. The C programming language is a standardized programming language developed in the early 1970s by Ken Thompson and Dennis Ritchie for use on the UNIX operating system. It has since spread to many other operating systems, and is one of the most widely used programming languages.

2. What are the types of constants in C?

In C language two types of constants are available:

- Primary constants
- Secondary constants

3. What are the types of C instructions?

There are basically three types of instructions in C:

- Type Declaration Instruction
- Arithmetic Instruction
- Control Instruction

4. What is a pointer?

Pointers are variables which stores the address of another variable. That variable may be a scalar (including another pointer), or an aggregate (array or structure). The pointed-to object may be part of a larger object, such as a field of a structure or an element in an array.

5. What is an array?

Array is a variable that hold multiple elements which has the same data type.

6. Differentiate between arrays and pointers?

Pointers are used to manipulate data using the address. Pointers use * operator to access the data pointed to by them.

Array is a collection of similar data type. Array use subscripted variables to access and manipulate data. Array variables can be equivalently written using pointer expression.

7. What is "this" pointer?

The "this" pointer is a pointer accessible only within the member functions of a class, struct, or union type. It points to the object for which the member function is called. Static member functions do not have a "this" pointer.

8. What are the uses of a pointer?

Pointer is used in the following cases

- It is used to access array elements.
- It is used for dynamic memory allocation.
- It is used in Call by reference.
- It is used in data structures like trees, graph, linked list etc.

9. What is the purpose of main() function?

The function main() invokes other functions within it. It is the first function to be called when the program starts execution.

- It is the starting function.
- It returns an int value to the environment that called the program.
- Recursive call is allowed for main() also.
- It is a user-defined function.

10. What are the different storage classes in C?

There are four types of storage classes in C language.

- Automatic
- Extern
- Register
- Static

11. What is a structure?

Structure constitutes a super data type which represents several different data types in a single unit. A structure can be initialized if it is static or global.

12. Define Constructors?

A constructor is a member function with the same name as its class. The constructor is invoked whenever an object of its associated class is created. It is called constructor because it constructs the values of data members of the class.

13. Define destructors?

A destructor is called for a class object when that object passes out of scope or is explicitly deleted. A destructors as the name implies is used to destroy the objects that have been created by a constructors. Like a constructor, the destructor is a member function whose name is the same as the class name but is preceded by a tilde.

14. What is the use of default constructor?

A constructor that accepts no parameters is called the default constructor. If no user-defined constructor exists for a class A and one is needed, the compiler implicitly declares a default parameter less constructor A::A(). This constructor is an inline public member of its class. The compiler will implicitly define A::A() when the compiler uses this constructor to create an object of type A. The constructor will have no constructor initializer and a null body.

15. What is a macro?

Macros are the identifiers that represent statements or expressions.

16. What is the difference between #include< > and #include " " ?

#include< > ----> specifically used for built in header files.

#include " " ----> Specifically used for user defined/created header file.

17. What are the advantages of the functions?

- It reduces the Complexity in a program by reducing the code.
- Functions are easily understanding and reliability and execution is faster.
- It also reduces the Time to run a program. In other way, it's directly proportional to Complexity.
- It's easy to find-out the errors due to the blocks made as function definition outside the main function.

18. How do declare an array?

We can declare an array by specify its data type, name and the number of elements the array holds between square brackets immediately following the array name.

syntax :

```
data_type array_name[size];
```

19. What are the differences between structures and union?

A structure variable contains each of the named members, and its size is large enough to hold all the members. Structure elements are of same size.

A Union contains one of the named members at a given time and is large enough to hold the largest member. Union element can be of different sizes.

20. What is the difference between an Array and a List?

The main difference between an array and a list is how they internally store the data whereas Array is collection of homogeneous elements. List is collection of heterogeneous elements.

21. What is the difference between a string copy (strcpy) and a memory copy (memcpy)?

The strcpy() function is designed to work exclusively with strings. It copies each byte of the source string to the destination string and stops when the terminating null character () has been moved.

On the other hand, the memcpy() function is designed to work with any type of data. Because not all data ends with a null character, you must provide the memcpy() function with the number of bytes you want to copy from the source to the destination.

22. What is the difference between const char*p and char const* p?

const char*p - p is pointer to the constant character. i.e value in that address location is constant.

const char* const p - p is the constant pointer which points to the constant string, both value and address are constants.

23. What is the purpose of realloc()??

Realloc(ptr,n) function uses two arguments.

- The first argument ptr is a pointer to a block of memory for which the size is to be altered.
- The second argument n specifies the new size. The size may be increased or decreased.

24. What is the use of typedef?

The typedef help in easier modification when the programs are ported to another machine. A descriptive new name given to the existing data type may be easier to understand the code.

25. What are the differences between new and malloc?

- New initializes the allocated memory by calling the constructor. Memory allocated with new should be released with delete.
- Malloc allocates uninitialized memory.
- The allocated memory has to be released with free. New automatically calls the constructor while malloc(doesn't)

26. What is the difference between strdup and strcpy?

Both copy a string. strcpy wants a buffer to copy the string. strdup allocates a buffer using malloc(). Unlike strcpy(), strdup() is not specified by ANSI.

27. What is this pointer?

It is a pointer that points to the current object. This can be used to access the members of the current object with the help of the arrow operator.

28. What is recursion?

A recursion function is one which calls itself either directly or indirectly it must halt at a definite point to avoid infinite recursion.

29. What are the characteristics of arrays in C?

- An array holds elements that have the same data type.
- Array elements are stored in subsequent memory locations
- Two-dimensional array elements are stored row by row in subsequent memory locations.
- Array name represents the address of the starting element

30. Differentiate between for loop and a while loop? What are it uses?

For executing a set of statements fixed number of times we use for loop while when the number of iterations to be performed is not known in advance we use while loop.

31. What is the difference between printf(...) and sprintf(...)?

- printf(...) -----> is standard output statement
- sprintf(.....)-----> is formatted output statement.

32. What is an explicit constructor?

A conversion constructor declared with the explicit keyword. The compiler does not use an explicit constructor to implement an implied conversion of types. Explicit constructors are simply constructors that cannot take part in an implicit conversion.

33. What is copy constructor?

Copy constructor is a constructor function with the same name as the class and used to make deep copy of objects.

34. What is the difference between malloc and calloc?

- Malloc() is use for memory allocation and initialize garbage values. malloc() for allocating the single block of memory.
- Calloc() is same as malloc() but it initialize 0 value. calloc() for allocating multiple blocks of memory.

35. What is null pointer?

NULL pointer is a pointer which is pointing to nothing.

Examples :

```
int *ptr=(char *)0;
float *ptr=(float *)0;
```

36. What is dynamic array?

The dynamic array is an array data structure which can be resized during runtime which means elements can be added and removed.

37. What are macros? What are its advantages and disadvantages?

Macros are abbreviations for lengthy and frequently used statements. When a macro is called the entire code is substituted by a single line though the macro definition is of several lines.

The advantage of macro is that it reduces the time taken for control transfer as in case of function. The disadvantage of it is here the entire code is substituted so the program becomes lengthy if a macro is called several times.

38. What are register variables? What are the advantages of using register variables?

If a variable is declared with a register storage class, it is known as register variable. The register variable is stored in the CPU register instead of main memory. Frequently used variables are declared as register variable as it's access time is faster.

39. What is storage class? What are the different storage classes in C?

Storage class is an attribute that changes the behavior of a variable. It controls the lifetime,scope and linkage. The storage classes in c are auto, register, and extern, static, typedef.

40. What the advantages of using Unions?

When the C compiler is allocating memory for unions it will always reserve enough room for the largest member.

41. In C, why is the void pointer useful? When would you use it?

The void pointer is useful because it is a generic pointer that any pointer can be cast into and back again without loss of information.

42. What is the difference between the functions memmove() and memcpy()?

The arguments of memmove() can overlap in memory. The arguments of memcpy() cannot.

43. Can a Structure contain a Pointer to itself?

Yes such structures are called self-referential structures.

44. What is dynamic memory allocation?

A dynamic memory allocation uses functions such as malloc() or calloc() to get memory dynamically. If these functions are used to get memory dynamically and the values returned by these function are assigned to pointer variables, such a way of allocating memory at run time is known as dynamic memory allocation.

45. What is pointer to a pointer?

If a pointer variable points another pointer value. Such a situation is known as a pointer to a pointer.

Example :

```
int *p1, **p2,
v=10;
P1=&v;
p2=&p1;
```

Here p2 is a pointer to a pointer.

46. What is a function?

A large program is subdivided into a number of smaller programs or subprograms. Each subprogram specifies one or more actions to be performed for the larger program. Such sub programs are called functions.

47. What is an argument?

An argument is an entity used to pass data from the calling to a called function.

48. What is the difference between syntax vs logical error?

Syntax Error

- These involve validation of syntax of language.
- Compiler prints diagnostic message.

Logical Error

- Logical error are caused by an incorrect algorithm or by a statement mistyped in such a way that it doesn't violate syntax of language.
- Difficult to find.

49. Explain enumerated types.

- Enumerated types allow the programmers to use more meaningful words as values to a variable.
- Each item in the enumerated type variable is actually associated with a numeric code.

50. Differentiate between the expression "++a" and "a++"?

- With ++a, the increment happens first on variable a, and the resulting value is used. This is called as prefix increment.
- With a++, the current value of the variable will be used in an operation. This is called as postfix increment.