



ANIL NEERUKONDA INSTITUTE OF TECHNOLOGY & SCIENCES

Autonomous status accorded by UGC and Andhra University

Approved by AICTE, Permanently Affiliated to Andhra University

Accredited by NBA (IT,CSE,EEE,ECE, and Mech) & accredited by NAAC with "A" Grade

Sangivalasa - 531162, Bheemunipatnam (Mandal), Visakhapatnam (Dist.)

Phone: 08933 - 225083, 225084, 226131, Fax: 08933-226395

Email: principal@anits.edu.in

COLLEGE CODE - ANIL



DEPARTMENT OF COMPUTER SCIENCES&ENGINEERING

VISION:

Our vision is to emerge as a world class Computer Science and Engineering department through excellent teaching and strong research environment that responds swiftly to the challenges of changing computer science technology and addresses technological needs of the stakeholders.

MISSION:

To enable our students to master the fundamental principles of computing and to develop in them the skills needed to solve practical problems using contemporary computer-based technologies and practices to cultivate a community of professionals who will serve the public as resources on state-of- the-art computing science and information technology.

Course outcomes:

After completion of this course, a student will be able to:

- | | |
|----|---|
| 1. | Develop C programs using operators |
| 2. | Write C programs using conditional structures |
| 3. | Write C programs using iterative structure arrays and strings |
| 4. | Inscribe C programs that use Pointers to and functions |
| 5. | Develop a c program for implementing user defined types and file processing |

PROGRAM OUTCOMES (POs):

Graduate Attribute1:	Engineering Knowledge
PO-A	An ability to apply the knowledge of basic engineering sciences, humanities, core engineering and computing concept in modeling and designing computer based systems.
Graduate Attribute2:	Problem Analysis
PO-B	An ability to identify, analyze the problems in different domains and define the requirements appropriate to the solution.
Graduate Attribute3:	Design/Development of Solution
PO-C	An ability to design, implement & test a computer based system, component or process that meet functional constraints such as public health and safety, cultural, societal and environmental considerations.
Graduate Attribute4:	Conduct Investigations of Complex Problems
PO-D	An ability to apply computing knowledge to conduct experiments and solve complex problems, to analyze and interpret the results obtained within specified timeframe and financial constraints consistently.
Graduate Attribute5:	Modern Tool Usage
PO-E	An ability to apply or create modern techniques and tools to solve engineering problems that demonstrate cognition of limitations involved in design choices.
Graduate Attribute6:	The Engineer and Society
PO-F	An ability to apply contextual reason and assess the local and global impact of professional engineering practices on individuals, organizations and society.
Graduate Attribute7:	Environment and Sustainability
PO-G	An ability to assess the impact of engineering practices on societal and environmental sustainability.
Graduate Attribute8:	Ethics
PO-H	Ability to apply professional ethical practices and transform into good responsible citizens with social concern.

Graduate Attribute9:	Individual and Team Work
PO-I	Acquire capacity to understand and solve problems pertaining to various fields of engineering and be able to function effectively as an individual and as a member or leader in a team.
Graduate Attribute10:	Communication
PO-J	An ability to communicate effectively with range of audiences in both oral and written forms through technical papers, seminars, presentations, assignments, project reports etc.
Graduate Attribute11:	Project Management and Finance
PO-K	An ability to apply the knowledge of engineering, management and financial principles to develop and critically assess projects and their outcomes in multidisciplinary areas.
Graduate Attribute12:	Life-long Learning
PO-L	An ability to recognize the need and prepare oneself for lifelong self learning to be abreast with rapidly changing technology.

PROGRAM SPECIFIC OUTCOMES (PSOs):

1.Programming and software Development skills: Ability to acquire programming efficiency to analyze, design and develop optimal solutions, apply standard practices in software project development to deliver quality software product.

2.Computer Science Specific Skills: Ability to formulate, simulate and use knowledge in various domains like data engineering, image processing and information and network security, artificial intelligence etc., and provide solutions to new ideas and innovations.

**ANIL NEERUKONDA INSTITUTE OF TECHNOLOGY
AND SCIENCES**

A Laboratory Manual
For
PROBLEM SOLVING WITH C (CSE-117)
SEMESTER-1



Prepared by
Mrs.S.A.BHAVANI
Assistant Professor
Dept of CSE

DEPARTMENT OF COMPUTER SCIENCE &ENGINEERING

Sl.No	List of Experiments	CO									
1	<p><u>1. CONVERTING MILES TO KILOMETERS</u></p> <p>PROBLEM STATEMENT: Your summer surveying job requires you to study some maps that give distances in kilometers and some that use miles. You and your coworkers prefer to deal in metric measurements. Write a program that performs the necessary conversion.</p> <p>Problem Input: miles /* the distance in miles*/ Problem Output: kms /* the distance in kilometers */ Relevant Formula: 1 mile = 1.609 kilometers</p> <p>Design algorithm , flow chart ,program using the above data requirements for the given problem. Try the sample test cases given below :</p> <table border="1" style="margin-left: auto; margin-right: auto;"> <thead> <tr> <th style="text-align: center;">SAMPLE TEST CASES</th> <th style="text-align: center;">INPU T</th> <th style="text-align: center;">OUPU T</th> </tr> </thead> <tbody> <tr> <td style="text-align: center;">Test case 1</td> <td style="text-align: center;">10</td> <td style="text-align: center;">16.09</td> </tr> <tr> <td style="text-align: center;">Test case 2</td> <td style="text-align: center;">2</td> <td style="text-align: center;">3.218</td> </tr> </tbody> </table>	SAMPLE TEST CASES	INPU T	OUPU T	Test case 1	10	16.09	Test case 2	2	3.218	1
SAMPLE TEST CASES	INPU T	OUPU T									
Test case 1	10	16.09									
Test case 2	2	3.218									
2	<p><u>SUPERMARKET COIN PROCESSOR</u></p> <p>PROBLEM STATEMENT : You are drafting software for the machines placed at the front of supermarkets to convert change to personalized credit slips. In this draft, the user will manually enter the number of each kind of coin in the collection, but in the final version, these counts will be provided by code that interfaces with the counting devices in the machine.</p> <p>Problem Inputs char first, middle, last /* a customer's initials */ int dollars /* number of dollars */ int quarters /* number of quarters */ int dimes /* number of dimes */ int nickels /* number of nickels */ int pennies /* number of pennies */</p> <p>Problem Outputs int total_dollars /* total dollar value */ int change /* leftover change */</p> <p>Additional Program Variables int total_cents /* total value in cents */</p> <p>Design algorithm, flow chart ,program using the above data requirements for the given problem Try the sample test cases given below :</p>	1									

TESTING TIP :

To test this program, try running it with a combination of coins that yield an exact dollar amount with no leftover change. For example, 1 dollar, 8 quarters, 0 dimes, 35 nickels, and 25 pennies should yield a value of 5 dollars and 0 cents. Then increase and decrease the quantity of pennies by 1 (26 and 24 pennies) to make sure that these cases are also handled properly.

SAMPLE TEST CASES	INPUT
Test case 1	Type in your 3 initials and press re JRH, please enter your coin inform Number of \$ coins > 2 Number of quarters> 14 Number of dimes > 12 Number of nickels > 25 Number of pennies > 131
Test case 2	Type in your 3 initials and press re JRH, please enter your coin inform Number of \$ coins > 3 Number of quarters> 12 Number of dimes > 14 Number of nickels > 50 Number of pennies > 175

3

WATER BILL PROBLEM

PROBLEM STATEMENT :Write a program that computes a customer's water bill. The bill includes a \$35 water demand charge plus a consumption (use) charge of \$1.10 for every thousand gallons used. Consumption is figured from meter readings (in thousands of gallons) taken recently and at the end of the previous quarter. If the customer's unpaid balance is greater than zero, a \$2 late charge is assessed as well.

Problem Constants

```
DEMAND_CHG 35.00 /* basic water demand charge
*/
PER_1000_CHG 1.10 /* charge per thousand gallons
used */
LATE_CHG 2.00 /* surcharge on an unpaid balance
*/
```

Problem Inputs

```
int previous /* meter reading from previous quarter in
thousands of gallons */
int current /* meter reading from current quarter */
double unpaid /* unpaid balance of previous bill */
```

Problem Outputs

```
double bill /* water bill */
double use_charge /* charge for actual water use */
```

1,2

double late_charge /* charge for nonpayment of part of previous balance */

Relevant Formulas

water bill = demand charge + use charge + unpaid balance+ applicable late charge

Design algorithm , flow chart ,program using the above data requirements for the given problem Try the sample test cases given below :

SAMPLE TEST CASES	INPUT
Test case 1	This program figures a water bill based on the demand charge (\$35.00) and a \$1.10 per 1000 gallons use charge. A \$2.00 surcharge is added to account with an unpaid balance. Enter unpaid balance, previous and current meter readings on separate lines after the prompts. Press <return> or <enter> after typing each number Enter unpaid balance> \$71.50 Enter previous meter reading> 419 Enter current meter reading> 4238
Test case 2	This program figures a water bill based on the demand charge (\$35.00) and a \$1.10 per 1000 gallons use charge. A \$2.00 surcharge is added to account with an unpaid balance. Enter unpaid balance, previous and current meter readings on separate lines after the prompts. Press <return> or <enter> after typing each number Enter unpaid balance> \$51 Enter previous meter reading> 419 Enter current meter reading> 4137

4

PRIME NUMBER

PROBLEM STATEMENT :Given a positive integer N, calculate the sum of all prime numbers between 1 and N (inclusive).

Input:

The first line of input contains an integer T denoting the number of test cases. T testcases follow. Each test case contains one line of input containing N.

Output:

For each test case, in a new line, print the sum of all prime numbers between 1 and N.

Constraints:

1 ≤ T ≤ 100
 1 ≤ N ≤ 10⁶

1,2

	<p>Design algorithm , flow chart ,program using the above data requirements for the given problem Try the sample test cases given below :</p> <table border="1" data-bbox="469 327 1233 629"> <thead> <tr> <th>SAMPLE TEST CASES</th> <th>INPUT</th> </tr> </thead> <tbody> <tr> <td>Test case 1</td> <td>2 5 10</td> </tr> <tr> <td>Test case 2</td> <td>2 7 10</td> </tr> </tbody> </table>	SAMPLE TEST CASES	INPUT	Test case 1	2 5 10	Test case 2	2 7 10	
SAMPLE TEST CASES	INPUT							
Test case 1	2 5 10							
Test case 2	2 7 10							
5	<p>BUBBLE SORT PROBLEM STATEMENT :The task is to complete bubble function which is used to implement Bubble Sort Input: First line of the input denotes the number of test cases 'T'. First line of the test case is the size of array and second line consists of array elements. Output: Sorted array in increasing order is displayed to the user. Constraints: 1 <=T<= 100 1 <=N<= 1000 1 <=arr[i]<= 1000 Design algorithm , flow chart ,program using the above data requirements for the given problem Try the sample test cases given below :</p> <table border="1" data-bbox="469 1308 1233 1644"> <thead> <tr> <th>SAMPLE TEST CASES</th> <th>INPUT</th> </tr> </thead> <tbody> <tr> <td>Test case 1</td> <td>2 5 4 1 3 9 7 10 10 9 8 7 6 5 4 3 2 1</td> </tr> <tr> <td>Test case 2</td> <td>1 5 8 9 3 2 0</td> </tr> </tbody> </table>	SAMPLE TEST CASES	INPUT	Test case 1	2 5 4 1 3 9 7 10 10 9 8 7 6 5 4 3 2 1	Test case 2	1 5 8 9 3 2 0	3
SAMPLE TEST CASES	INPUT							
Test case 1	2 5 4 1 3 9 7 10 10 9 8 7 6 5 4 3 2 1							
Test case 2	1 5 8 9 3 2 0							
6	<p>TEXT EDITOR PROBLEM STATEMENT: Design and implement a program to perform editing operations on a line of text. Your editor should be able to locate a specified target substring, delete a substring, and insert a substring at a specified location. The editor should expect source strings of less than 80 characters.</p> <p>Problem Constant MAX_LEN 100 /* maximum size of a string */ Problem Inputs char source[MAX_LEN] /* source string */ char command /* edit command */ Problem Output</p>	3						

	<p>char source[MAX_LEN] /* modified source string */ Design algorithm , flow chart ,program using the above data requirements for the given problem Try the sample test cases given below :</p> <table border="1" data-bbox="469 439 1235 817"> <thead> <tr> <th data-bbox="469 439 788 517">SAMPLE TEST CASES</th> <th data-bbox="788 439 1235 517">INPUT</th> </tr> </thead> <tbody> <tr> <td data-bbox="469 517 788 703">Test case 1</td> <td data-bbox="788 517 1235 703">Enter the source string: > Internet use is growing rapid Enter D(Delete), I(Insert), F(Find) or Q(Quit)> d String to delete> growing</td> </tr> <tr> <td data-bbox="469 703 788 817">Test case 2</td> <td data-bbox="788 703 1235 817">Enter D(Delete), I(Insert), F(Find) or Q(Quit)> F String to find> .</td> </tr> </tbody> </table>	SAMPLE TEST CASES	INPUT	Test case 1	Enter the source string: > Internet use is growing rapid Enter D(Delete), I(Insert), F(Find) or Q(Quit)> d String to delete> growing	Test case 2	Enter D(Delete), I(Insert), F(Find) or Q(Quit)> F String to find> .	
SAMPLE TEST CASES	INPUT							
Test case 1	Enter the source string: > Internet use is growing rapid Enter D(Delete), I(Insert), F(Find) or Q(Quit)> d String to delete> growing							
Test case 2	Enter D(Delete), I(Insert), F(Find) or Q(Quit)> F String to find> .							
7	<p>ARITHMETIC WITH COMMON FRACTIONS PROBLEM STATEMENT: You are working problems in which you must display your results as integer ratios; therefore, you need to be able to perform computations with common fractions and get results that are common fractions in reduced form. You want to write a program that will allow you to add, subtract, multiply, and divide several pairs of common fractions.</p> <p>Design algorithm, flow chart, program using the above data requirements for the given problem Try the sample test cases given below :</p> <table border="1" data-bbox="469 1308 1235 1756"> <thead> <tr> <th data-bbox="469 1308 788 1346">SAMPLE TEST CASES</th> <th data-bbox="788 1308 1235 1346">INPUT 1</th> </tr> </thead> <tbody> <tr> <td data-bbox="469 1346 788 1447">Test case 1</td> <td data-bbox="788 1346 1235 1447">Enter a common fraction as two integers separated by a slash> 3/-4</td> </tr> <tr> <td data-bbox="469 1447 788 1756">Test case 2</td> <td data-bbox="788 1447 1235 1756">Enter a common fraction as two integers separated by a slash> 3/4 Enter an arithmetic operator (+,-,*,/) > + Enter a common fraction as two integers separated by a slash> 5/8 Entering find_gcd with n1 = 44, n2 = 55 Do another problem? (y/n)>n</td> </tr> </tbody> </table>	SAMPLE TEST CASES	INPUT 1	Test case 1	Enter a common fraction as two integers separated by a slash> 3/-4	Test case 2	Enter a common fraction as two integers separated by a slash> 3/4 Enter an arithmetic operator (+,-,*,/) > + Enter a common fraction as two integers separated by a slash> 5/8 Entering find_gcd with n1 = 44, n2 = 55 Do another problem? (y/n)>n	1,2,3
SAMPLE TEST CASES	INPUT 1							
Test case 1	Enter a common fraction as two integers separated by a slash> 3/-4							
Test case 2	Enter a common fraction as two integers separated by a slash> 3/4 Enter an arithmetic operator (+,-,*,/) > + Enter a common fraction as two integers separated by a slash> 5/8 Entering find_gcd with n1 = 44, n2 = 55 Do another problem? (y/n)>n							
8	<p>FACTORIAL OF A NUMBER PROBLEM STATEMENT : Find factorial of a given number n. Design algorithm , flow chart ,program using the above data requirements for the given problem Try the sample test cases given below :</p> <table border="1" data-bbox="469 2018 1235 2168"> <thead> <tr> <th data-bbox="469 2018 788 2096">SAMPLE TEST CASES</th> <th data-bbox="788 2018 1235 2096">INPUT 1</th> </tr> </thead> <tbody> <tr> <td data-bbox="469 2096 788 2134">Test case 1</td> <td data-bbox="788 2096 1235 2134">Enter a number to find factorial</td> </tr> <tr> <td data-bbox="469 2134 788 2168">Test case 2</td> <td data-bbox="788 2134 1235 2168">Enter a number to find factorial</td> </tr> </tbody> </table>	SAMPLE TEST CASES	INPUT 1	Test case 1	Enter a number to find factorial	Test case 2	Enter a number to find factorial	4
SAMPLE TEST CASES	INPUT 1							
Test case 1	Enter a number to find factorial							
Test case 2	Enter a number to find factorial							

9	<p>COLLECTING AREA FOR SOLAR-HEATED HOUSE – FILES AND FUNCTIONS</p> <p>PROBLEM STATEMENT : An architect needs a program that can estimate the appropriate size for the collecting area of a solar-heated house. Determining collecting area size requires consideration of several factors, including the average number of heating degree days for the coldest month of a year (the product of the average difference between inside and outside temperatures and the number of days in the month), the heating requirement per square foot of floor space, the floor space, and the efficiency of the collection method. The program will have access to two data files. File hdd.txt contains numbers representing the average heating degree days in the construction location for each of 12 months. File solar.txt contains the average solar insolation (rate in BTU/day at which solar radiation falls on one square foot of a given location) for each month. The first entry in each file represents data for January, the second, data for February, and so on.</p> <p>Problem Inputs Average heating degree days file Average solar insolation file heat_deg_days /* average heating degree days for coldest month */ coldest_mon /* coldest month (number 1 .. 12) */ solar_insol /* average daily solar insolation (BTU/ft^2) for coldest month */ heating_req /* BTU/degree day ft^2 for planned type construction*/ efficiency /* % of solar insolation converted to usable heat */ floor_space /* square feet */</p> <p>Program Variables energy_resrc /* usable solar energy available in coldest month (BTUs obtained from 1 ft^2 of collecting area) */</p> <p>Problem Outputs heat_loss /* BTUs of heat lost by structure in coldest month */ collect_area /* approximate size (ft^2) of collecting area needed*/</p> <p>The formula for approximating the desired collecting area (A) is :</p> <p>A= heat loss / energy resource Design algorithm , flow chart ,program using the above data requirements for the given problem Try the sample test cases given below :</p>	4

	SAMPLE TEST CASES	INPUT	OUTPUT
	Test case 1	What is the approximate heating requirement (BTU / degree day ft ²) of this type of construction? =>9 What percent of solar insolation will be converted to usable heat? => 60 What is the floor space (ft ²)? => 1200	To replace heat loss of 11350800 B in the coldest month (month 12) with available solar insolation of 500 BT ft ² / day, and an efficiency of 60 percent, use a solar collecting area of 1221 ft ² .
	Test case 2	What is the approximate heating requirement (BTU / degree day ft ²) of this type of construction? =>10 What percent of solar insolation will be converted to usable heat? => 60 What is the floor space (ft ²)? => 1200	To replace heat loss of 12612000 B in the coldest month (month 12) with available solar insolation of 500 BT ft ² / day, and an efficiency of 60 percent, use a solar collecting area of 1221 ft ² .
10	<p>UNIVERSAL MEASUREMENT CONVERSION</p> <p>PROBLEM STATEMENT: Design a program that takes a measurement in one unit (e.g., 4.5 quarts) and converts it to another unit (e.g., liters). For example, this conversion request 450 km miles would result in this program output Attempting conversion of 450.0000 km to miles . . . 450.0000km = 279.6247 miles . The program should produce an error message if a conversion between two units of different classes (e.g., liquid volume to distance) is requested. The program should take a database of conversion information from an input file before accepting conversion problems entered interactively by the user. The user should be able to specify units either by name (e.g., kilograms) or by abbreviation (e.g., kg).</p> <p>Structured Data Type</p> <p>unit_t members :</p> <pre> name /* character string such as "milligrams" */ abbrev /* shorter character string such as "mg" */ class /* character string "liquid_volume", "distance", or "mass" */ standard /* number of standard units that are equivalent to this unit */ </pre> <p>Problem Constants</p> <pre> NAME_LEN 30 /* storage allocated for a unit name */ ABBREV_LEN 15 /* storage allocated for a unit abbreviation */ CLASS_LEN 20 /* storage allocated for a measurement class */ </pre>		4,5

```

MAX_UNITS 20 /* maximum number of different
units handled */
Problem Inputs
unit_t units[MAX_UNITS] /* array representing unit
conversion factors database */
double quantity /* value to convert */
char old_units[NAME_LEN] /* name or abbreviation of
units to be converted */
char new_units[NAME_LEN] /* name or abbreviation of
units to convert to */

```

Problem Output

Message giving conversion.

Data file units.txt:

```

miles      mi      distance  1609.3
kilometers km      distance  1000
yards      yd      distance  0.9144
meters     m       distance  1
quarts     qt      liquid_volume 0.94635
liters     l       liquid_volume 1
gallons    gal     liquid_volume 3.7854
milliliters ml     liquid_volume 0.001
kilograms  kg      mass      1
grams      g       mass      0.001
slugs      slugs   mass      0.14594
pounds     lb      mass      0.43592

```

Design algorithm , flow chart ,program using the above data requirements for the given problem

Try the sample test cases given below :

SAMPLE TEST CASES	INPUT 1
Test case 1	Enter a conversion problem or q to quit. To convert 25 kilometers to miles, you would enter > 25 kilometers miles or, alternatively, > 25 km mi
Test case 2	Enter a conversion problem or q to quit. > 2.5 qt l Attempting conversion of 2.5000 qt to l ... 2.5000qt = 2.3659 l Enter a conversion problem or q to quit.

LIST OF INDUSTRY RELEVANT SKILLS:

- Proficiency with programming languages. ...
- Learning concepts and applying them to other problems. ...

- Mathematical skills. ...
- Problem-solving capability. ...
- Communication skills. ...
- Writing skills. ...
- Inquisitiveness. ...
- Self-motivation.

GUIDELINES TO TEACHERS:

- Faculty must verify the observations and records before assign the system.
- Faculty must verify Students Id cards before enter into Lab
- Faculty must take the attendance starting and ending of the lab time period.

This lab course consists of two set of programs

- 1) Minimum set of sample programs
- 2) Additional set of programs

Minimum set of sample programs are designed unit wise covering all the topics in the theory .
Additional set of programs are designed basing on problem solving

Sessional marks : 50 marks

- 1) Daily Evaluation (Includes Record, Observation & regular performance) – 30 marks
- 2) Attendance – 5 marks
- 3) Internal Exam – 10 marks
- 4) Viva Voce – 5 marks

Daily Evaluation (30 marks)

Every Student must execute minimum set of sample programs to secure 60% of marks in Daily Evaluation i.e. 18 Marks and to appear in external examination.

In addition to that if a student finishes the minimum set and 5 programs from additional set of programs would secure 80% of marks in Daily Evaluation i.e. 24 Marks.

If a student finishes all the programs in both the set s will secure 100% of marks in Daily Evaluation

Internal Exam (10 marks)

- Every student is given 4 questions in the internal exam out of which the difficulty level of 2 questions is easy / medium and 2 questions of difficulty level is high
- Each easy / medium level question carries 20% of marks and difficulty level question carries 30% of marks

External Exam (50 marks)

- Viva voce – 10 marks
- Write up + Execution – 40 marks

Write up + Execution (40 marks)

- Every student is given 4 questions in the external exam out of which the difficulty level of 2 questions is easy / medium and 2 questions of difficulty level is high
- Each easy / medium level question carries 30% of marks and difficulty level question carries 20% of marks.

INSTRUCTIONS TO STUDENTS:

- Students should use computer related components smoothly

- Students should not carry other items into lab.
- Students must wear the dress code and ID cards.
- Every student is given 4 questions in the external exam out of which the difficulty level of 2 questions is easy / medium and 2 questions of difficulty level is high
- Each easy / medium level question carries 30% of marks and difficulty level question carries 20% of marks.

GUIDELINES TO LAB PROGRAMMERS:

- Lab Programmers must verify All the Systems whether they are working properly or not.
- Lab Programmers must verify All the other equipment's(devices like ACs).

LAB RUBRICS:

Key Performance Criteria(KPC) (25 pts)	4-Very Good	3-Good	2-Fair	1-Need to improve
Problem Statement (2)	Detail understanding of the problem (2)	Understanding of the problem (2)	Basic understanding of the problem (1)	Partial understanding of the problem (1)
Experimental Procedure/ algorithm/ flow chart/ analysis (4)	The procedure is explained and well designed the problem with appropriate analysis (4)	The procedure is explained and designed the problem with analysis (3)	Missing some experimental procedure with partial analysis (2)	Missing major experimental details and analysis (1)
Implementation (4)	Implement Optimal solution with appropriate results for all the inputs	Implement solution with correct results for most of the inputs	implement solution with the correct answers for some inputs and results wrong answers for some cases	Implement Solution does not produce the appropriate results for the given inputs
Test Case verification (3)	Produces correct output for all possible test cases(3)	Produces correct output for most of the test cases (2)	Produces correct output for some of the test cases (2)	Produces Wrong output for most of the test cases (1)
Viva voice / oral presentation(5)	In depth knowledge on the concept and answered all the questions(5)	Good knowledge on the concept and answered all the questions(4)	Basic knowledge on the concept and answered some of the questions(3)	With basic knowledge on the concept and answered few questions(2)
Presentation of record / documentation(4)	Presented the content effectively and Submitted on time (4)	Presented the content and Submitted on time (3)	Presented the in-complete content and Submitted . (2)	Presented the wrong content and submitted in delay.(1)
Code of conduct (courtesy, safety, behavioral aspects, ethics etc.)(3)	While conducting the procedure, the student is in proper dress code, always respectful of others and leaves the area clean.(3)	While conducting the procedure, the student is in proper dress code, many times respectful of others and leaves the area clean only after being reminded.(2)	While conducting the procedure, the student is in partial dress code, sometimes respectful of others and leaves the area clean only after being reminded.(2)	While conducting the procedure, the student is not in proper dress code , not respectful of others and leaves the area messy even after being reminded.(1)

PRACTICAL 1: CONVERTING MILES TO KILOMETERS :

1. Practical significance :

1. Write comment to make your programs readable.
2. Use descriptive variables in your programs (Name of the variables should show their purposes)

2. Relevant Program Outcomes :

PO1, PO2, PO3, PO4

3. Competency and practical skills :

Students can able to test applications by passing the parameter

4. Prerequisites :

Mathematical formulas

5. Resources required :

Software Requirements (Recommend)	Operating System : Windows7/Linux/Ubuntu Application Software : Java SE 9 Text Editor : Notepad
Hardware Requirements (Recommend)	Monitor : 15" 1024x768 LCD Ram : 4GB Harddrive: 512GB

6. Precautions:

- Check Whether the computer is getting proper power or not.
- Ensure the keyboard, mouse and monitor are properly working.
- Ensure that there are no power fluctuations while executing the commands.
- Safe working conditions help prevent injury to people and damage to computer equipment.
- A safe work space is clean, organized, and properly lighted. Everyone must understand and follow safety procedures.
- Follow electrical safety guidelines to prevent electrical fires, injuries, and fatalities in the home and the workplace.
- Power supplies and CRT monitors contain high voltage.

7. Algorithm/circuit/Diagram/Description:

Step 1: start

Step 2: read no of miles.

Step 3: convert miles to kilometers by multiplying 1.609.

Step 4: print distance in kilometers.

Step 5: stop

8. Test cases: 2

9. Sample output:

Sample test cases	Input	Output
Test case 1	10	16.09
Test case 2	2	3.218

10. Practical Related Questions:

1. what is operators?

2. type of operators?

11 .Exercise Questions :

1. Write a C program to generate a random number

2. Write program to convert months to days to hours to seconds?

2. SUPERMARKET COIN PROCESSOR

1. Practical significance :

1. Write comment to make your programs readable.

2. Use descriptive variables in your programs (Name of the variables should show their purposes)

2. Relevant Program Outcomes :

PO1, PO2, PO3, PO4

3. Competency and practical skills :

Students can able to test applications by passing the parameter

4. Prerequisites :

Mathematical formulas and currency details about different countries.

5. Resources required :

Software Requirements (Recommend)	Operating System : Windows7/Linux/Ubuntu Application Software : Java SE 9 Text Editor : Notepad
Hardware Requirements (Recommend)	Monitor : 15" 1024x768 LCD Ram : 4GB Harddrive: 512GB

6. Precautions:

- Check Whether the computer is getting proper power or not.
- Ensure the keyboard, mouse and monitor are properly working.
- Ensure that there are no power fluctuations while executing the commands.
- Safe working conditions help prevent injury to people and damage to computer equipment.
- A safe work space is clean, organized, and properly lighted. Everyone must understand and follow safety procedures.
- Follow electrical safety guidelines to prevent electrical fires, injuries, and fatalities in the home and the workplace.
- Power supplies and CRT monitors contain high voltage.

7. Algorithm/circuit/Diagram/Description:

Step 1: Start.

Step 2: Enter inputs for dollars, quarters, dimes, nickels, pennies

Step 3: $\text{total_cents} = (\text{dollars} * 100) + (\text{quarters} * 25) + (\text{dimes} * 10) + (\text{nickels} * 5) + (\text{pennies})$

Step 4: $\text{total_dollars} = (\text{total_cents}) / 100$.

Step 5: $\text{Change} = (\text{total_cents}) - (\text{total_dollars} * 100)$.

- Step 6: Print total_dollars.
- Step 7: print change.
- Step 8: Stop.

8. Test cases:

9. Sample output:

SAMPLE CASES	TEST	INPUT	OUPUT
Test case 1		Type in your 3 initials and press return> JRH JRH, please enter your coin information. Number of \$ coins > 2 Number of quarters> 14 Number of dimes > 12 Number of nickels > 25 Number of pennies > 131	JRH Coin Credit Dollars: 9 Change: 26 cents
Test case 2		Type in your 3 initials and press return> JRH JRH, please enter your coin information. Number of \$ coins > 3 Number of quarters> 12 Number of dimes > 14 Number of nickels > 50 Number of pennies > 175	JRH Coin Credit Dollars: 11 Change: 26 cents

10. Practical Related Questions:

1. What are the basic data types associated with C?
2. What are reserved words with a programming language?

11 .Exercise Questions :

1. Write a C program to accept a coordinate point in a XY coordinate system and determine in which quadrant the coordinate point lies.
2. Write a C program to find the eligibility of admission for a professional course based on the following criteria: [Go to the editor](#)
Eligibility Criteria : Marks in Maths ≥ 65 and Marks in Phy ≥ 55 and Marks in Chem ≥ 50 and Total in all three subject ≥ 190 or Total in Maths and Physics ≥ 140
----- Input the marks obtained in Physics :65 Input the marks obtained in Chemistry :51
Input the marks obtained in Mathematics :72 Total marks of Maths, Physics and Chemistry : 188 Total marks of Maths and Physics : 137 The candidate is not eligible.
Expected Output :
The candidate is not eligible for admission.

3. WATER BILL PROBLEM

1. Practicalsignificance :

1. Write comment to make your programs readable.
2. Use descriptive variables in your programs (Name of the variables should show their purposes)

2.Relevant Program Outcomes :

PO1,PO2,PO3,PO4

3.Competency and practical skills :

Students can able to test applications by passing the parameter.

4.Prerequisites :

Mathematical formulas and knowledge on oparetors.

5.Resources required :

Software Requirements (Recommend)	Operating System : Windows7/Linux/Ubuntu Application Software : Java SE 9 Text Editor : Notepad
Hardware Requirements (Recommend)	Monitor : 15" 1024x768 LCD Ram : 4GB Harddrive: 512GB

6.Precautions:

- Check Whether the computer is getting proper power or not.
- Ensure the keyboard, mouse and monitor are properly working.
- Ensure that there are no power fluctuations while executing the commands.
- Safe working conditions help prevent injury to people and damage to computer equipment.
- A safe work space is clean, organized, and properly lighted. Everyone must understand and follow safety procedures.
- Follow electrical safety guidelines to prevent electrical fires, injuries, and fatalities in the home and the workplace.
- Power supplies and CRT monitors contain high voltage.

7.Algorithm/circuit/Diagram/Description:

Step 1: Start.

Step 2:Enter inputs for current reading , previous reading,unpaid_balance.

Step 3: total_due=(current reading- previous reading)1.10+unpaid_balance+35.

Step 4:if(unpaid_balance>0)

then print "bill includes \$2.00 late charge on unpaid_balance"

total_due+=2

print total_due

else

print "total_due"

8. Test cases:

9.Sample output:

SAMPLE TEST CASES	INPUT	OUPUT
Test case 1	This program figures a water bill based on the demand charge (\$35.00) and a \$1.10 per 1000 gallons use charge. A \$2.00 surcharge is added to accounts with an unpaid balance.	Bill includes \$2.00 late charge on unpaid balance of \$71.50 Total due = \$152.50

	<p>Enter unpaid balance, previous and current meter readings on separate lines after the prompts. Press <return> or <enter> after typing each number.</p> <p>Enter unpaid balance> \$71.50 Enter previous meter reading> 4198 Enter current meter reading> 4238</p>	
Test case 2	<p>This program figures a water bill based on the demand charge (\$35.00) and a \$1.10 per 1000 gallons use charge.</p> <p>A \$2.00 surcharge is added to accounts with an unpaid balance.</p> <p>Enter unpaid balance, previous and current meter readings on separate lines after the prompts. Press <return> or <enter> after typing each number.</p> <p>Enter unpaid balance> \$51 Enter previous meter reading> 4198 Enter current meter reading> 4137</p>	<p>Bill includes \$2.00 late charge on unpaid balance of \$71.50 Total due = \$102.00</p>

10. Practical Related Questions:

1. Difference between design making statements and switch statements?

11 .Exercise Questions :

1. Write a C program to read roll no, name and marks of three subjects and calculate the total, percentage and division.
2. Write a C program to accept the height of a person in centimeter and categorize the person according to their height

4. Prime Numbers between 1 to N:

1. Practicallsignificance :

1. Write comment to make your programs readable.
2. Use descriptive variables in your programs (Name of the variables should show their purposes)

2. Relevant Program Outcomes :

PO1, PO2, PO3, PO4

3. Competency and practical skills :

Students can able to test applications by passing the parameter

4. Prerequisites :

Declarations and definitions of variables, functions, loops and definition of Prime Number.

5. Resources required :

Software Requirements (Recommend)	Operating System : Windows7/Linux/Ubuntu Application Software : Java SE 9 Text Editor : Notepad
Hardware Requirements (Recommend)	Monitor : 15" 1024x768 LCD Ram : 4GB Harddrive: 512GB

6.Precautions:

- Check Whether the computer is getting proper power or not.
- Ensure the keyboard, mouse and monitor are properly working.
- Ensure that there are no power fluctuations while executing the commands.
- Safe working conditions help prevent injury to people and damage to computer equipment.
- A safe work space is clean, organized, and properly lighted. Everyone must understand and follow safety procedures.
- Follow electrical safety guidelines to prevent electrical fires, injuries, and fatalities in the home and the workplace.
- Power supplies and CRT monitors contain high voltage.

7.Algorithm/circuit/Diagram/Description:

Algorithm Sieve of Eratosthenes is

input: an integer $n > 1$.

output: all prime numbers from 2 through n .

let A be an **array of Boolean** values, indexed by **integers** 2 to n , initially all **set to true**.

for $i = 2, 3, 4, \dots$, not exceeding \sqrt{n} **do**

if $A[i]$ **is true**

for $j = i^2, i^2+i, i^2+2i, i^2+3i, \dots$, not exceeding n **do**

$A[j] := \text{false}$

return all i such that $A[i]$ **is true**

8. Test cases: 9.Sample output:

SAMPLE TEST CASES	INPUT	OUPUT
Test case 1	2 5 10	10 17
Test case 2	2 7 10	17 17

10.Practical Related Questions:

- 1.What is the time complexity of algorithm used by you?
2. What is the best algorithm you can use to implement this program?

11 .Exercise Questions :

1. Write a C program to accept two integers and check whether they are equal or not.?

5. Bubble Sort:

1. Practical significance :

1. Write comment to make your programs readable.

2. Use descriptive variables in your programs (Name of the variables should show their purposes)

2. Relevant Program Outcomes :

PO1, PO2, PO3, PO4

3. Competency and practical skills :

Students can be able to test applications by passing the parameter.

A) Information about the topic in brief. i.e. how sorting works.

B) Formulas related to the topic/s (Swapping of two numbers is required).

C) Work out few examples and dry run before implementing the actual code.

4. Prerequisites :

Declarations and definitions of variables, functions, loops and concept of sorting, swapping.

5. Resources required :

Software Requirements (Recommend)	Operating System : Windows7/Linux/Ubuntu Application Software : Java SE 9 Text Editor : Notepad
Hardware Requirements (Recommend)	Monitor : 15" 1024x768 LCD Ram : 4GB Harddrive: 512GB

6. Precautions:

- Check Whether the computer is getting proper power or not.
- Ensure the keyboard, mouse and monitor are properly working.
- Ensure that there are no power fluctuations while executing the commands.
- Safe working conditions help prevent injury to people and damage to computer equipment.
- A safe work space is clean, organized, and properly lighted. Everyone must understand and follow safety procedures.
- Follow electrical safety guidelines to prevent electrical fires, injuries, and fatalities in the home and the workplace.
- Power supplies and CRT monitors contain high voltage.

7. Algorithm/circuit/Diagram/Description:

Bubble Sort:

```
beginBubbleSort(list)
  for all elements of list
    if list[i] > list[i+1]
      swap(list[i], list[i+1])
    end if
  end for
return list
end BubbleSort
```

8. Test cases: 9.Sample output:

SAMPLE TEST CASES	INPUT	OUPUT
Test case 1	2 5 4 1 3 9 7 10 10 9 8 7 6 5 4 3 2 1	1 3 4 7 9 1 2 3 4 5 6 7 8 9 10
Test case 2	1 5 8 9 3 2 0	0 2 3 8 9

10.Practical Related Questions:

- 1.What is the time complexity of algorithm used by you?
2. What is the best algorithm you can use to implement this program?
3. Explain how merge sort works?
4. Explain Quick sort?
5. Why Quick Sort is preferred than Merge sort in most of the applications?

11.Exercise Questions :

- 1.Write a program in C to count the total number of words in a string.
- 2.Write a program in C to count total number of alphabets, digits and special characters in a string

6.Text Editor:

1. Practicalseignificance :

Translating given algorithm to valid program.

2.Relevant Program Outcomes :

PO1,PO2,PO3,PO4

3.Competency and practical skills :

Students can able to test applications by passing the parameter

4.Prerequisites :

Problem Constant MAX_LEN 100 /* maximum size of a string */

Problem Inputs

char source[MAX_LEN] /* source string */

char command /* edit command */

Problem Output

char source[MAX_LEN] /* modified source string */

5.Resources required :

Software Requirements (Recommend)	Operating System : Windows7/Linux/Ubuntu Application Software : Java SE 9 Text Editor : Notepad
Hardware Requirements (Recommend)	Monitor : 15" 1024x768 LCD Ram : 4GB Harddrive: 512GB

6.Precautions:

- Check Whether the computer is getting proper power or not.
- Ensure the keyboard, mouse and monitor are properly working.
- Ensure that there are no power fluctuations while executing the commands.
- Safe working conditions help prevent injury to people and damage to computer equipment.
- A safe work space is clean, organized, and properly lighted. Everyone must understand and follow safety procedures.
- Follow electrical safety guidelines to prevent electrical fires, injuries, and fatalities in the home and the workplace. Power supplies and CRT monitors contain high voltage

7.Algorithm/circuit/Diagram/Description:

- 1.Enter the source string.
- 2.create a menu bar using control structure staments.

8. Test cases:

9.Sample output:

SAMPLE TEST CASES	INPUT	OUPUT
Test case 1	Enter the source string: > Internet use is growing rapidly. Enter D(Delete), I(Insert), F(Find), or Q(Quit)> d String to delete> growing	New source: Internet use is rapidly
Test case 2	Enter D(Delete), I(Insert), F(Find), or Q(Quit)> F String to find> .	'.' found at position 23

10.Practical Related Questions:

2. Define memmove?
- 1.Reverse words in a given string without string functions
3. Define strstr?
4. Define strtok?
5. Define strcmp?

11 .Exercise Questions :

1. Write a program in C to count a total number of duplicate elements in an array

7. ARITHMETIC WITH COMMON FRACTIONS

1. Practical significance :

Translating given algorithm to valid program.

2.Relevant Program Outcomes :

PO1,PO2,PO3,PO4

3.Competency and practical skills :

Students can able to test applications by passing the parameter

4.Prerequisites :

- Check Whether the computer is getting proper power or not.
- Ensure the keyboard, mouse and monitor are properly working.
- Ensure that there are no power fluctuations while executing the commands.
- Safe working conditions help prevent injury to people and damage to computer equipment.

- A safe work space is clean, organized, and properly lighted. Everyone must understand and follow safety procedures.
- Follow electrical safety guidelines to prevent electrical fires, injuries, and fatalities in the home and the workplace.
- Power supplies and CRT monitors contain high voltage.

5.Resources required :

Software Requirements (Recommend)	Operating System : Windows7/Linux/Ubuntu Application Software : Java SE 9 Text Editor : Notepad
Hardware Requirements (Recommend)	Monitor : 15" 1024x768 LCD Ram : 4GB Harddrive: 512GB

6.Precautions:

- Check Whether the computer is getting proper power or not.
- Ensure the keyboard, mouse and monitor are properly working.
- Ensure that there are no power fluctuations while executing the commands.
- Safe working conditions help prevent injury to people and damage to computer equipment.
- A safe work space is clean, organized, and properly lighted. Everyone must understand and follow safety procedures.
- Follow electrical safety guidelines to prevent electrical fires, injuries, and fatalities in the home and the workplace. Power supplies and CRT monitors contain high voltage.

7.Algorithm/circuit/Diagram/Description:

```
/* calculating the numerator */
x3 = (x1 * y2) + (x2 * y1);
```

```
/* calculating the denominator */
y3 = (y1 * y2);
```

```
/* simplifying the fraction */
if (x3 > y3)
    div = y3;
else
    div = x3;
for (i = div; i > 0; i--)
    if (x3 % i == 0 && y3 % i == 0) {
        x3 = x3 / i;
        y3 = y3 / i;
    }
```

8. Test cases: 9.Sample output.

SAMPLE TEST CASES	INPUT 1	OUPUT
Test case 1	Enter a common fraction as two integers separated by a slash> 3/-4	Input invalid— denominator must be positive
Test case 2	Enter a common fraction as two integers separated by a slash> 3/4 Enter an arithmetic operator (+,-,*, or /) > + Enter a common fraction as two integers separated by a slash> 5/8 Entering find_gcd with n1 = 44, n2 = 32 Do another problem? (y/n)>n	gcd of 44 and 32?> 4 find_gcd returning 4 3/4 + 5/8 = 11/8

10. Practical Related Questions:

1. What is modulus operator?
2. Define complex number?
3. Difference b/w "/" & "%" operator?

11. Exercise Questions :

1. C program to print all natural numbers upto N without using semi-colon

8. TO FIND THE FACTORIAL OF A GIVEN NUMBER N

1. Practical significance :

Find out the value of the factorial number.
How to use operators and loops.

2. Relevant Program Outcomes :

PO1, PO2, PO3, PO

3. Competency and practical skills :

Students can able to test applications by passing the parameter

4. Prerequisites :

The factorial of a number is the product of all the integers from 1 to that number.

For example, the factorial of 6 (denoted as 6!) is $1*2*3*4*5*6 = 720$.

Factorial is not defined for negative numbers and the factorial of zero is one, $0! = 1$.

5. Resources required :

Software Requirements (Recommend)	Operating System : Windows7/Linux/Ubuntu Application Software : Java SE 9 Text Editor : Notepad
Hardware Requirements (Recommend)	Monitor : 15" 1024x768 LCD Ram : 4GB Harddrive: 512GB

6. Precautions:

- Check Whether the computer is getting proper power or not.
- Ensure the keyboard, mouse and monitor are properly working.
- Ensure that there are no power fluctuations while executing the commands.
- Safe working conditions help prevent injury to people and damage to computer equipment.
- A safe work space is clean, organized, and properly lighted. Everyone must understand and follow safety procedures.
- Follow electrical safety guidelines to prevent electrical fires, injuries, and fatalities in the home and the workplace. Power supplies and CRT monitors contain high voltage.

7. Algorithm/circuit/Diagram/Description:

- step 1. Start
- step 2. Read the number n
- step 3. [Initialize]
i=1, fact=1

- step 4. Repeat step 4 through 6 until $i=n$
- step 5. $fact=fact*i$
- step 6. $i=i+1$
- step 7. Print fact
- step 8. Stop

8. Test cases: 9.Sample output:

SAMPLE TEST CASES	INPUT 1	OUPUT
Test case 1	Enter a number to find factorial>2	Factorial of 2 is 2
Test case 2	Enter a number to find factorial>3	Factorial of 3 is 6

10.Practical Related Questions:

11 .Exercise Questions :

1. C program to print sum of two integers witht out +

9. COLLECTING AREA FOR SOLAR-HEATED HOUSE – FILES AND FUNCTIONS

1. Practicalsignificance :

How to use functions and multi way selection.
Develop C program using functions and conditional control statements.

2.Relevant Program Outcomes :

PO1,PO2,PO3,PO4

3.Competency and practical skills :

Students can able to test applications by passing the parameter

4.Prerequisites :

The formula for approximating the desired collecting area (A) is $A=heat\ loss/energy\ resource$
In turn, heat loss is computed as the product of the heating requirement, the floor space, and the heating degree days. We compute the necessary energy resource by multiplying the efficiency of the collection method by the average solar insulation per day and the number of days.
In this program we will use three input sources: the two data files and the keyboard. We can now identify the problem's data requirements and develop an algorithm.

5.Resources required :

Software Requirements (Recommend)	Operating System : Windows7/Linux/Ubuntu Application Software : Java SE 9 Text Editor : Notepad
Hardware Requirements (Recommend)	Monitor : 15" 1024x768 LCD Ram : 4GB Harddrive: 512GB

6.Precautions:

- Check Whether the computer is getting proper power or not.

- Ensure the keyboard, mouse and monitor are properly working.
- Ensure that there are no power fluctuations while executing the commands.
- Safe working conditions help prevent injury to people and damage to computer equipment.
- A safe work space is clean, organized, and properly lighted. Everyone must understand and follow safety procedures.
- Follow electrical safety guidelines to prevent electrical fires, injuries, and fatalities in the home and the workplace. Power supplies and CRT monitors contain high voltage

7. Algorithm/circuit/Diagram/Description:

- Step 1. Determine the coldest month and the average heating degree days for this month.
 Step 2. Find the average daily solar insolation per ft² for the coldest month.
 Step 3. Get from the user the other problem inputs: heating_req , efficiency , and floor_space .
 Step 4. Estimate the collecting area needed.
 Step 5. Display results.

STEP 1 REFINEMENT

We will introduce three new variables to use in our refinement—a counter, ct, to keep track of our position in the heating degree days file, an integer variable to record file status, and an integer variable next_hdd to hold each heating degree days value in turn.

Additional Program Variables

```
ct          /* position in file */
status      /* input status */
next_hdd    /* one heating degree days value */
```

- 1.1 Scan first value from heating degree days file into heat_deg_days , and initialize coldest_mon to 1.
- 1.2 Initialize ct to 2.
- 1.3 Scan a value from the file into next_hdd , saving status .
- 1.4 As long as no faulty data or not at end of file, repeat
- 1.5 if next_hdd is greater than heat_deg_days
- 1.6 Copy next_hdd into heat_deg_days .
- 1.7 Copy ct into coldest_mon .
- 1.8 Increment ct .
- 1.9 Scan a value from the file into next_hdd , saving status .

STEP 4 REFINEMENTS

- 4.1 Calculate heat_loss as the product of heating_req , floor_space , and heat_deg_days .
 - 4.2 Calculate energy_resrc as the product of efficiency (converted to hundredths), solar_insol , and the number of days in the coldest month.
 - 4.3 Calculate collect_area as heat_loss divided by energy_resrc . Round result to nearest whole square foot.
- We will develop a separate function for finding the number of days in a month, a value needed in step 4.2

8. Test cases: 9. Sample output:

SAMPLE TEST CASES	INPUT	OUPUT
Test case 1	What is the approximate heating requirement (BTU / degree day ft ²) of this type of construction? =>9 What percent of solar insolation will be converted to usable heat? => 60 What is the floor space (ft ²)? => 1200	To replace heat loss of 11350800 BTU in the coldest month (month 12) with available solar insolation of 500 BTU / ft ² / day, and an efficiency of 60 percent, use a solar collecting area of 1221 ft ² .

10. Practical Related Questions:

1. what is the difference between formal arguments actual arguments?
2. categorie of functions?

11 .Exercise Questions :

1. Write a program in C to check whether a given number is a Kaprekar number or not.

10. UNIVERSAL MEASUREMENT CONVERSION:

1. Practicalsignificance :

How to use functions and multi way selection.

Develop C program using functions and conditional control statements.

2. Relevant Program Outcomes :

PO1,PO2,PO3,PO4

3. Competency and practical skills :

Students can able to test applications by passing the parameter

4. Prerequisites :

Structured Data Type

unit_t members :

```
name /* character string such as "milligrams" */
abbrev /* shorter character string such as "mg" */
class /* character string "liquid_volume", "distance", or "mass" */
standard /* number of standard units that are equivalent to this unit */
```

5. Resources required :

Software Requirements (Recommend)	Operating System : Windows7/Linux/Ubuntu Application Software : Java SE 9 Text Editor : Notepad
Hardware Requirements (Recommend)	Monitor : 15" 1024x768 LCD Ram : 4GB Harddrive: 512GB

6. Precautions:

- Check Whether the computer is getting proper power or not.
- Ensure the keyboard, mouse and monitor are properly working.
- Ensure that there are no power fluctuations while executing the commands.
- Safe working conditions help prevent injury to people and damage to computer equipment.
- A safe work space is clean, organized, and properly lighted. Everyone must understand and follow safety procedures.
- Follow electrical safety guidelines to prevent electrical fires, injuries, and fatalities in the home and the workplace. Power supplies and CRT monitors contain high

7. Algorithm/circuit/Diagram/Description:

8. Test cases: 9. Sample output

SAMPLE TEST CASES	INPUT 1	OUPUT
Test case 1	Enter a conversion problem or q to quit. To convert 25 kilometers to miles, you would enter > 25 kilometers miles or, alternatively, > 25 km mi	>450 km miles Attempting conversion of 450.0000 km to miles ... 450.0000km = 279.6247 miles
Test case 2	Enter a conversion problem or q to quit. > 2.5 qt l Attempting conversion of 2.5000 qt to l ... 2.5000qt = 2.3659 l Enter a conversion problem or q to quit.	> 100 meters gallons Attempting conversion of 100.0000 meters to gallons ... Cannot convert meters (distance) to gallons (liquid_volume)

10. Practical Related Questions:

1. what is the use of nested structure?
2. how to access structure member from structure?

11 .Exercise Questions :

1. Traffic light , traffic light simulation.

