Unified Civil Engineering Computation Tool (UCECT)

Requirements:-

Component Requirement:

Processor (CPU): Dual-Core Processor

RAM : 512 GB (OR) MORE

HDD : 20GB

Software Requirements:

Operating System

Windows : 7 / 8 / 10 / 11. OR Linux   OR macOS

Programming Tools:

C Compiler

Turbo C

Code Editors / IDE:

CodeBlocks

VS Code

Dev C++

Turbo C++

INTRODUCTION:

1. Unified Civil Engineering Computation Tool (UCECT) is a C-based software tool designed to perform multiple civil engineering calculations in one unified system.

2. It includes three core modules:

Stopping Sight Distance (SSD), Soil Bearing Capacity Checker and Water Tank Volume Calculator.

3. Stopping Sight Distance (SSD) Calculator:

This program helps in road curve design by calculating the minimum distance a vehicle needs to stop safely.
It takes user inputs such as vehicle speed, reaction time, friction factor and gradient and computes SSD using standard transportation engineering formulas.

4. Soil Bearing Capacity Checker:

This program checks whether a foundation is safe under the applied load.
It takes applied load, footing area, and safe bearing capacity (SBC) as inputs and determines if the actual soil pressure is within safe limits.

5. Water Tank Volume Calculator:

This program calculates the water storage capacity of different tank shapes (rectangular or cylindrical).
It takes user inputs such as length, breadth, height, radius, or diameter, and computes the tank volume in cubic meters and litres.

ALGORITHM:

Stopping Sight Distance (SSD)

Step 1: Start.

Step 2: Read input: Ask user to enter Speed V in km/h.

Step 3: Read input: Ask user to enter reaction time t in seconds (typical 2.5 s if unknown).

Step 4: Read input: Ask user to enter coefficient of friction f (dimensionless).

Step 5:Read input: Ask user to enter grade G as a decimal (e.g., +0.02 for +2% uphill, −0.03 for −3% downhill).

Step 6: Validate inputs: Ensure $V \geq 0$, $t > 0$, $f > 0$. If invalid show error and stop.

Step 7: Compute lag (thinking) distance LD:
$LD = 0.278 \times V \times t$.

Step 8: Compute braking denominator:
$denom = 254 \times (f + G)$.
If $denom \leq 0$, show error (unsafe/invalid friction+grade) and stop.

Step 9: Compute braking distance BD:
$BD = (V \times V) / denom$.

Step 10: Compute SSD:
$SSD = LD + BD$.

Step 11: Output: Display LD, BD, SSD with units (meters).

Step 12: End.

Soil Bearing Capacity Checker —

Step 1: Start.

Step 2: Read input: Ask user to enter applied total vertical load P in kN.

Step 3: Read input: Ask user to enter footing area A in m^2.

Step 4: Validate inputs: Ensure $P \geq 0$ and $A > 0$. If invalid show error and stop.

Step 5: Compute ultimate bearing pressure $q = P / A$ (kN/m^2).

Step 6: Apply factor of safety (optional): If user supplies FS (typical 2.5–3), compute allowable / safe bearing capacity SBC = q / FS. If no FS, assume SBC = q and note it is unfactored.

Step 7: Classify soil (example thresholds):
If SBC ≤ 100 kN/m^2 → “Weak soil”
If 100 < SBC ≤ 200 kN/m^2 → “Medium strength”
If SBC > 200 kN/m^2 → “Strong soil”
(Mention these are example thresholds — user may change them.)

Step 8: Output: Display q, SBC (if FS used), and classification. Include units (kN/m^2).

Step 9: End.

Water Tank Volume Calculator — (Menu + two shapes)

Step 1: Start.

Step 2: Display menu:

1. Rectangular tank
2. Cylindrical tank

Step 3: Read input: User selects choice (1 or 2).

Step 4:Validate choice: If invalid show error and return to menu or stop.

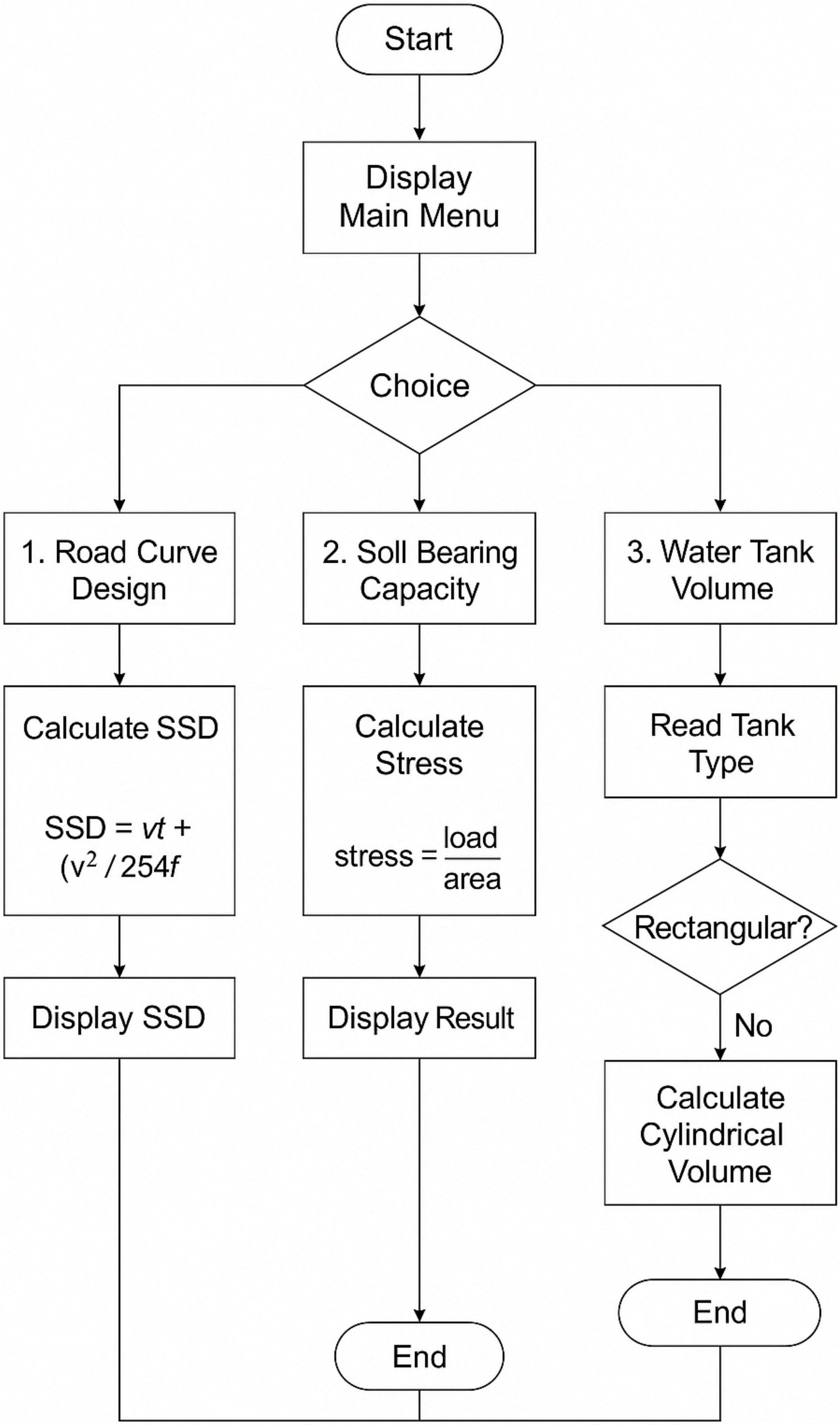
If choice = 1 (Rectangular):

5. Read length L (m).
6. Read width W (m).
7. Read height H (m).
8. Validate L>0, W>0, H>0 — if not, show error and stop.
9. Compute Volume $V = L \times W \times H$ (m^3).
10. Optionally compute litres = $V \times 1000$.
11. Display V (m^3) and litres.
12. End.

If choice = 2 (Cylindrical):

5. Read radius r (m) or diameter d (m). If diameter given, set $r = d/2$.
6. Read height h (m).
7. Validate r>0 and h>0 — if not, show error and stop.
8. Compute Volume $V = \pi \times r^2 \times h$ (use $\pi = 3.14159265$).
9. Optionally compute litres = $V \times 1000$.
10. Display V (m^3) and litres.
11. End.

FLOWCHART:



Start
Display Main Menu
Choice
1. Road Curve Design
2. Soll Bearing Capacity
3. Water Tank Volume
Calculate SSD
SSD = vt + (v2 / 254f
Calculate Stress
stress = load / area
Read Tank Type
Rectangular?
No
Display SSD
Display Result
Calculate Cylindrical Volume
End
End

MODULES for the Project:

1. Road Curve Design – Stopping Sight Distance (SSD)

Module 1: Input Module

Design speed

Brake reaction time

Coefficient of friction

Gradient

Module 2: SSD Calculation Module

Perception–reaction distance

Braking distance

Total Stopping Sight Distance (SSD)

Module 3: Output Module

Reaction distance

Braking distance

Final SSD result

Safe/unsafe indication for road design

2. Soil Bearing Capacity Checker

Module 1: Input Module

Soil type

Cohesion (c)

Angle of internal friction (ϕ)

Unit weight (γ)
Foundation width (B)
Foundation depth (D)

Module 2: Bearing Capacity Calculation Module

Terzaghi's formula
Calculates ultimate bearing capacity
Calculates net and safe bearing capacity (using FOS)

Module 3: Output Module

Ultimate bearing capacity
Net SBC
Safe SBC
Suitable foundation recommendation

3. Water Tank Volume Calculator

Module 1: Input Module

Select tank type (rectangular/cylindrical)
Rectangular: Length, Width, Height
Cylindrical: Radius, Height

Module 2: Volume Calculation Module

Volume in m^3
Converts to litres

Module 3: Output Module

Volume in cubic meters
Volume in litres

LOGIC (Module wise logic):

1. ROAD CURVE DESIGN – SSD

Module 1: Input Logic

1. Ask the user to enter design speed (V).
2. Ask for brake reaction time (t).
3. Ask for the coefficient of friction (f).
4. Ask for gradient value (G).
5. Store all inputs for calculation.

Module 2: SSD Calculation Logic

1. Convert speed V from km/hr to m/s → .
2. Calculate reaction distance → .
3. Calculate braking distance →

$d_2 = Vm^2 / (2 \times g \times (f \pm G))$

Use –G for uphill

4. Compute Total SSD → .
5. Store SSD value for output.

Module 3: Output Logic

1. Display reaction distance.
2. Display braking distance.
3. Display final SSD.
4. If SSD > minimum required → Print “Safe”.
5. Else → Print “Not Safe”.

2. SOIL BEARING CAPACITY CHECKER:

Module 1: Input Logic

1. Ask the user for soil type.
2. Input cohesion (c).
3. Input angle of internal friction (ϕ).
4. Input unit weight of soil (γ).

5. Input foundation width (B) and depth (D).
6. Store all values.

Module 2: Bearing Capacity Calculation Logic

1. Use Terzaghi's bearing capacity formula:

If $\phi > 0$:

$q_u = cN_c + \gamma DN_q + 0.5\gamma BN\gamma$

If $\phi = 0$ (Clay soil):

$q_u = 5.7c$

2. Calculate net bearing capacity:

$q_{net} = q_u - \gamma D$

3. Apply factor of safety (FOS):

$q_{safe} = q_{net}/FOS$

4. Store final SBC.

Module 3: Output Logic

1. Print ultimate bearing capacity (q_u).
2. Print net bearing capacity.
3. Print safe bearing capacity.
4. Give foundation suggestion:

If SBC < 100 kN/m² → "Use raft foundation".

If SBC ≥ 100 → "Isolated footing is suitable".

3. WATER TANK VOLUME CALCULATOR

Module 1: Input Logic

1. Ask user to select tank type:

Press 1 → Rectangular

Press 2 → Cylindrical

2. If rectangular:

Input length, width, height.

3. If cylindrical:

Input radius and height.

4. Store values.

Module 2: Volume Calculation Logic

1. If rectangular tank:

Volume = L × W × H

Volume = $\pi \times R^2 \times H$

Litres = Volume × 1000

Calculate filling time:

Time = Volume / (Flow Rate)

Module 3: Output Logic

1. Display geometry type.
2. Display volume in cubic meters (m^3).
3. Display capacity in litres.
4. If flow rate provided → Show filling time.

OUTPUT:

==== MAIN MENU ====

1. Road Curve Design – Stopping Sight Distance (SSD)

2. Soil Bearing Capacity Checker

3. Water Tank Volume Calculator

4. Exit

Enter your choice: 1

=== Road Curve Design – Stopping Sight Distance (SSD) ===

Enter speed of vehicle (km/hr): 80

Enter reaction time (sec): 5

Enter coefficient of friction: 10

Stopping Sight Distance (SSD) = 402.52 meters

Press Enter to return to Main Menu...

==== MAIN MENU ====

1. Road Curve Design – Stopping Sight Distance (SSD)
2. Soil Bearing Capacity Checker
3. Water Tank Volume Calculator
4. Exit

Enter your choice: 2

=== Soil Bearing Capacity Checker ===

Enter load on foundation (kN): 100

Enter area of foundation (sq.m): 9

Enter safe bearing capacity of soil (kN/sq.m): 5

Applied Stress = 11.11 kN/m2

Result: NOT SAFE — Increase foundation size.

Press Enter to return to Main Menu...

==== MAIN MENU ====

1. Road Curve Design – Stopping Sight Distance (SSD)
2. Soil Bearing Capacity Checker
3. Water Tank Volume Calculator
4. Exit

Enter your choice: 3

=== Water Tank Volume Calculator ===

1. Rectangular Tank
2. Cylindrical Tank

Enter type: 1

Enter length (m): 6

Enter width (m): 6

Enter height (m): 6

```
Volume of Rectangular Tank = 216.00 cubic meters

Press Enter to return to Main Menu...

==== MAIN MENU ====
1. Road Curve Design – Stopping Sight Distance (SSD)
2. Soil Bearing Capacity Checker
3. Water Tank Volume Calculator
4. Exit
Enter your choice: 3

=== Water Tank Volume Calculator ===
1. Rectangular Tank
2. Cylindrical Tank
Enter type: 2
Enter radius (m): 6
Enter height (m): 6

Volume of Cylindrical Tank = 678.59 cubic meters

Press Enter to return to Main Menu...

==== MAIN MENU ====
1. Road Curve Design – Stopping Sight Distance (SSD)
2. Soil Bearing Capacity Checker
3. Water Tank Volume Calculator
4. Exit
Enter your choice: 4

Exiting program...
```

CONCLUSION:

The Unified Civil Engineering Computation Tool (UCECT) integrates important civil engineering calculations into a single C-based application. It helps users easily perform Stopping Sight Distance, Soil Bearing Capacity, and Water Tank Volume calculations. The menu-driven approach makes the program simple and user-friendly. Accurate input handling and validation ensure reliable results. The modular structure improves clarity and allows easy modification or expansion in the future. UCECT bridges the gap between theoretical knowledge and practical engineering applications. It runs on basic hardware and standard C programming tools. Overall, the project is a useful learning and computation tool for civil engineering students.