Documentation

(Group 28)

**Group Information**

Project Name - LDU Factorization

Group Members- Aditya

Deepak Mandal

Pragati Chauhan

Shubham Kang

Suyash Yadav

Overview of Project

In this project we developed a web application which demonstrates factorization of a matrix into LDU(Lower, Diagonal, Upper) matrix. User can access the application through-

* Give Input for dimension
* Give input for matrix A
* Will compute the Upper triangular matrix along with all elimination matrix.
* Determine the Lower triangular matrix by calculating inverse of elimination multiplication matrices.
* Construct Diagonal Matrix.
* Will Verify the LDU factorization.

HTML Elements

* **Dimension of Matrix**-

The button (<button onclick="takeorder('showmatrix')">) allow user to give input for desired matrix dimension.

* **Upper Triangular Matrix (U)**:

Users can compute UUU by clicking <button onclick="Umatrix('showUmatrix')">.

In the div of id=”showUmatrix” all the elimination matrices will display.

* **Matrix (L)**:

Users can calculate LLL with <button onclick="lmatrix('showlmatrix')">.

The result is displayed in div with id="showlmatrix".

* **Updated Upper Matrix**:

In the div with id="UUmatrix", by clicking on <button onclick="UUmatrix('uumat')"> user will get the Updated Upper triangular matrix.

* **Diagonal Matrix:**

In the div with id="dmaat", by clicking on <button onclick="Dmatrix('dmat')"> allow user display diagonal matrix.

* Finally, user will compare A’= LDU’ and at last A=A’ will verify the result.

CSS Styling

**Body**

* **Font**: Uses Arial, sans-serif for clean, modern typography.
* **Margin**: Adds 20px spacing around the content.
* **Text Color**: Sets text to black (rgb(7, 5, 5)) for readability.
* **Background**: Includes a full-cover background image with specific positioning for a visually appealing layout.

**Headings**

* **Primary Heading (h1)**:
  + Centered alignment.
  + Black color with a large font size (50px) for emphasis.
  + 40px bottom margin to separate it from other elements.
* **Secondary Heading (h2)**:
  + Smaller size (x-large).
  + Dark color (#070b0e) for subtle contrast.

**Text and Divs**

* **Paragraphs**:
  + Font size: 20px for readability.
  + Line height: 1.6 for well-spaced text.
* **Div Sections**:
  + Applies to specific IDs (#inputfield, #showmatrix, etc.).
  + Features:
    - White background.
    - Rounded corners (8px).
    - Light shadow for a floating effect.
    - Centered horizontally using left: 33%.
    - Fixed width of 30% for a consistent layout.

**Buttons**

* **Default**:
  + Green background (#4CAF50), white text for contrast.
  + Rounded corners (5px) for a softer look.
  + Padding (10px 15px) for larger clickable areas.
* **Hover State**:
  + Changes background to a darker green (#358739) for feedback.
* **Focus State**:
  + Removes the default outline for a cleaner UI.

**Input Fields**

* **Style**:
  + Light gray border (#ccc) and rounded corners for smooth edges.
  + Font size: 16px for clear visibility.
  + Maximum width: 300px for compact but usable input boxes.
* **Focus Effect**:
  + Border and outline change to green (#4CAF50) for interactivity.

**Comparison Section**

* **Default Layout**:
  + Uses flexbox to align items side by side.
  + Each child div occupies 48% of the width for balance.
* **Responsive Layout**:
  + Switches to a vertical column layout on small screens.
  + Centers items and adds 100% width for flexibility.

**Responsive Design**

* + Buttons and input fields expand to fill the available width.
  + Div sections stack vertically for better usability.
  + Spacing adjustments for a clutter-free interface.

JAVA Script Logic

* **Matrix Formation**

In the function *multwomat(m,mm)*, we have initiated the result matrix & multiplied two matrices m and m, by assuming both matrices are square & will store the final result in mult.

* **Creating an Elementary Matrix**

*function makeEM(row, col, value), is creating an elementary function.*

* **Entering the input**

*function takeorder(id),* is allowing users to give input for a square matrix.

* **Creating U Matrix along with all Elimination Matrices**

*function Umatrix(id),* is core part of the code , where it is calculating the U factor of LDU factorization along with all elimination matrices.

* **Calculating L matrix**

*function lmatrix(id),* is calculating the L part of LDU factorization along with the calculation of inverse of multiplication of all elimination matrices.

* **Generating U’ Matrix**

*function UUmatrix(uumat)*, is deriving U’ from U matrix , by ensuring that each row of U′ has a leading 1 in the pivot position.

* **Calculating D Matrix**

*function Dmatrix(dmat),* is responsible for constructing the diagonal matrix D from the upper triangular matrix U in the LDU factorization process.

* **Multiplication of all three factors**

*function multiplyLDU(LDU*), is multiplying all three factors i.e. L,D & U and verifying the factorization.

The application successfully demonstrates LDU factorization by decomposing a matrix A and reconstructing it through multiplication of L, D, and U′.