

DATA STRUCTURES AND ITS LIMITATIONS

John Jones¹ and Steve Smith²

¹ Assistant Professor, Department of Electrical Engineering, Bharati Vidyapeeth Deemed University, College of Engineering, Pune, India

² Assistant Professor, Department of Electrical Engineering, Bharati Vidyapeeth Deemed University, College of Engineering, Pune, India

Abstract

Abstract – In this growing age of technology, computers play one of the most important roles. Be it medical surgeries to handling of huge power grids and power generation stations, all are controlled by computers. The most basic requirement of a computer to carry out its task is data. The data that is provided is initially in the raw form, but in order for it to work with other machinery that is required to carry out the task and produce an output the data needs to be structured. This structuring of data is done by some simple commands and algorithms termed as data structure. [1] This review paper aims at introducing different types of data structures as well as addressing the shortcomings of certain data structure. I will also try to point out certain improvements that can be carried out in certain data structures.

Keywords: Keyword1, Keyword2, Keyword3

1 GENERAL GUIDELINES

Please read these instructions carefully. The objective of this template is to enable you in an easy way to style your article attractively in a style similar to that of the typeset journal. It should be emphasized, however, that the final appearance of your paper in print and in electronic media may likely vary to some extent from the presentation achieved in this template.

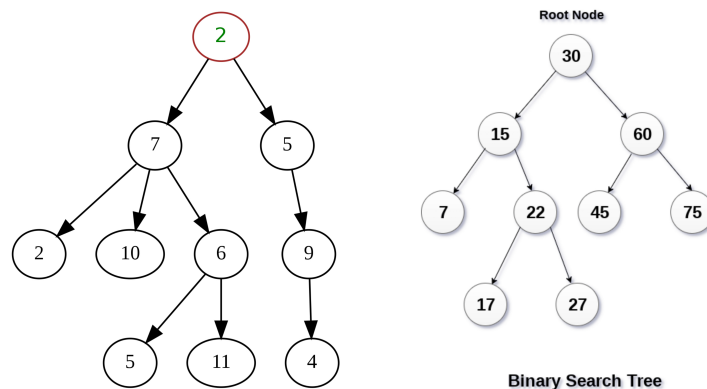


Figure 1. That was tree figure

1.1 Level Headings: Subsections

You will usually want to divide your article into (numbered) sections and subsections (perhaps even subsubsections). Use to help organize your document as appropriate.

1.2 Paragraphs

Paragraphs are **not** indented and a line break is included between paragraphs. The first paragraph in a section or subsection will automatically not have an indentation. Use `\noindent` as needed.

1.3 Hyperlinks / URLs

Include hyperlinks using URL `http://www.google.com`.

2 MATHEMATICS

Here is some mathematics. For $A \in M_n$, the factorization $A = LU$, where L is unit lower triangular and U is upper triangular, is called the *LU decomposition*, or *LU factorization*. We can use such a factorization, when it exists, to solve the system $Ax = b$ by first solving for the vector y in $Ly = b$ and then solving $Ux = y$. However, not every $n \times n$ matrix A has an LU decomposition. The following theorem provides conditions for the existence and uniqueness of an LU decomposition of a $n \times n$ matrix. A proof can be found in Johnson and Horn (1985, p. 160). An equation is given by Strang (1993),

$$x = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a} \quad (1)$$

Example 1. The 3×3 matrix $A = \begin{bmatrix} 1 & 5 & 1 \\ 1 & 4 & 2 \\ 4 & 10 & 2 \end{bmatrix}$ has all non-zero principle minors, A_1 , A_2 and A_3 .

Therefore, there is a unique LU factorization with both L and U nonsingular given by

$$\begin{bmatrix} 1 & 5 & 1 \\ 1 & 4 & 2 \\ 4 & 10 & 2 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ 1 & 1 & 0 \\ 4 & 10 & 1 \end{bmatrix} \begin{bmatrix} 1 & 5 & 1 \\ 0 & -1 & 1 \\ 0 & 0 & 12 \end{bmatrix}.$$

3 DEFINITION, THEOREM, COROLLARY

Definition 1. *Definitions are if and only if statements.*

```
\begin{definition}
Definitions are if and only if statements.
\end{definition}
```

Theorem 2 (Matrices with Infinitely Many LU Factorizations). *For $A \in M_n$, if two or more of any first $(n - 1)$ columns are linearly dependent or any of the first $(n - 1)$ columns are 0, then A has infinitely many LU factorizations.*

Proof. We will prove only for the the case when $A \in M_3$.

$$dm + r = e \Rightarrow r = e - dm \quad (2)$$

$$dn + rp = f \Rightarrow p = \frac{f - dn}{r} \quad (3)$$

$$gm + s = h \Rightarrow s = h - gm \quad (4)$$

$$gn + sp + t = i \Rightarrow t = i - sp - gn \quad (5)$$

□

Corollary 3. *If x , then y .*

4 EXAMPLES

Here is an example of an example.

Example 2. *Let $\{1, 2, 3\}$ and $\{2, 1, 3\}$ be two lists of integers. Then, to check if the two lists are equal we would have,*

$$\{1, 2, 3\} == \{2, 1, 3\} .$$

```
\begin{example}
Let  $\{1, 2, 3\}$  and  $\{2, 1, 3\}$  be two lists of integers. Then, to check
\begin{center}
\texttt{\{1, 2, 3\} == \{2, 1, 3\}}.
\end{center}\label{ex:equallists}
\end{example}
```

5 LISTS

NAGJ uses the `outline` package.

5.1 Enumerated List

The following code produces an enumerated list. The enumerated list numbers each list item with Arabic numerals.

```
\begin{outline}[enumerate]
  \1 First Level
    \2 Second level
      \3 Third level
\end{outline}
```

1. First Level

(a). Second level

i. Third level

5.2 Itemized List

The following code produces an itemized list.

```
\begin{outline}
  \1 First item
    \2 Second level item
      \3 Third level sub-item
\end{outline}
```

- First Level
 - Second level
 - * Third level

6 TABLES AND FIGURES

A **caption** that briefly describes the material is required for figures and tables. Any required information, such as photo credits or data source, may be included in captions. Furthermore, as part of the authorization to use that content, providers of this material may need a specific credit line, which might be placed in the caption (or wherever the provider has requested).

6.1 Figures

Figures should be high quality (1200 dpi for line art, 600 dpi for grayscale and 300 dpi for color, at the correct size). The preferred method of including graphics in the *North American GeoGebra Journal* is to export to TiKz. For exposition on TiKz, see (Quinlan, 2013). Other acceptable file formats include: EPS, PS, PNG, JPEG, or TIFF.

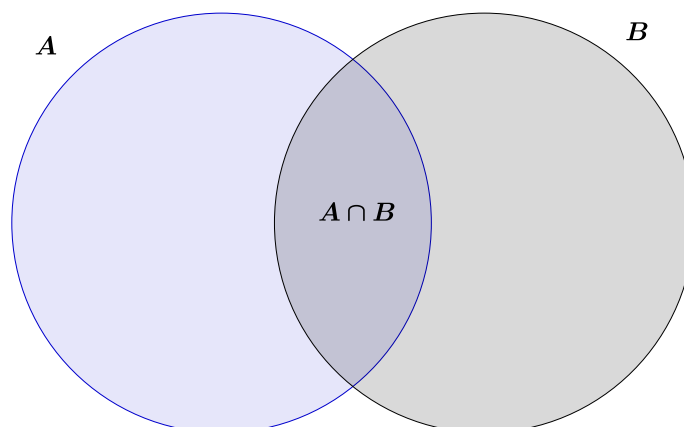


Figure 2. Provide a short caption description.

6.2 Tables

We use the `booktabs` package. Tables should present new information rather than duplicating what is in the text. Readers should be able to interpret the table without reference to the text. Please supply editable files.

Date	Time	Average	Standard Deviation
Jan 1	1100	4.7	0.6
Jan 2	2300	16.7	2.9
Jan 3	1400	11.4	3.5
Jan 4	1130	8.4	2.1
Jan 5	500	5.2	1.9
Jan 6	1700	7.9	2.2

Table 1. This is a caption to the table

7 CROSS REFERENCE

Figures, tables, and equations should be labeled (`label`) then referenced using the `ref`.

8 CITATIONS

Citations follow APA format. Johnson and Horn (1985) shows an example using `\citet{key}`, whereas to include citations parenthetically as in, (Johnson and Horn, 1985), use `\citep{key}`. You can also include page number of reference such as, (Johnson and Horn, 1985, p. 17), by using options in the cite command, i.e., `\citep[p. 17]{key}`. Multiple researchers (Johnson and Horn, 1985; Strang, 1993) can be cited using one cite command; in particular, with keys separated by commas `\citep{johnson1985matrix, strang1993introduction}`.

9 GEOGEBRA COMMAND LINE

Commands that could be entered at the GeoGebra command prompt should use the `verbatim` environment and can be centered.

$$J = (6 - (16/3 - \text{distance}(D, K)), y(C)) .$$

For inline commands such as `distance(A, B)`, use `\texttt` or `\verb`.

APPENDIX

Appendixes, if needed, appear before the acknowledgment.

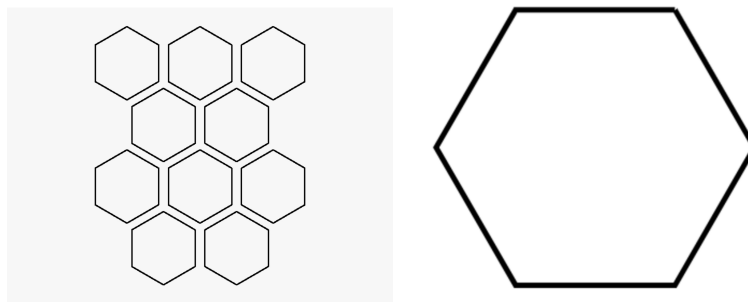


Figure 3. That was Hex Image

ACKNOWLEDGMENT

The preferred spelling of the word “acknowledgment” in American English is without an “e” after the “g.” Use the singular heading even if you have many acknowledgments. Avoid expressions such as “One of us (S.B.A.) would like to thank” Instead, write “F. A. Author thanks” In most cases, sponsor and financial support acknowledgments are placed in the unnumbered footnote on the first page, not here.

REFERENCES

- Johnson, C. R. and Horn, R. A. (1985). *Matrix analysis*. Cambridge University Press Cambridge.
- Quinlan, J. (2013). Geogebra as a frontend to generating graphics for latex. *North American GeoGebra Journal*, 2(1).
- Strang, G. (1993). *Introduction to linear algebra*, volume 3. Wellesley-Cambridge Press Wellesley, MA.



Author is an Associate Professor at all the best universities.



Author, is a professor and has taught you everything you know.

APPENDIX - CODE

```
for(int i = 0; i < dim - 1; i++){
    for(int k = i+1; k < dim; k++){
        for(int m = i+1; m < dim; m++){
            if(upper[i][i] == 0 && upper[m][i] != 0 ){
                'THERE IS NO LU FACTORIZATION'
            }
        }
        if(upper[i][i] != 0 && upper[k][i] != 0){
            multiplier =upper[i][i]/upper[k][i];
            lower[k][i] = multiplier;
        }else{
            if(upper[i][i] == 0 && upper[k][i] == 0){
                INFINITELY MANY LU FACTORIZATIONS
                multiplier = INPUT FROM USER;
                lower[k][i] = multiplier;
            }
        }
        for(int j = 0; j < dim; j++){
            row[j] = upper[i][j]*multiplier;
        }
        for(int r = 0; r < dim; r++){
            upper[k][r] = upper[k][r] - row[r];
        }
    }
}
Display Lower And Upper
(Perform Matrix Multiplication on L and U)
Display L and U
}
```