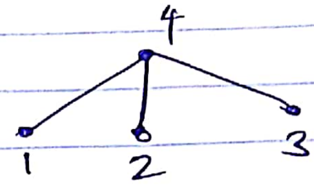
Seidel

Normal access

$$A \rightarrow \begin{bmatrix} 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 \\ 1 & 1 & 1 & 0 \end{bmatrix}$$



$A \rightarrow$ adjacency matrix

$i, j \rightarrow$ vertices

$D_{i,j} \rightarrow$ shortest distance between vertices i & j

$$A[1][1] = \begin{bmatrix} 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 \\ 1 & 1 & 1 & 0 \end{bmatrix}$$

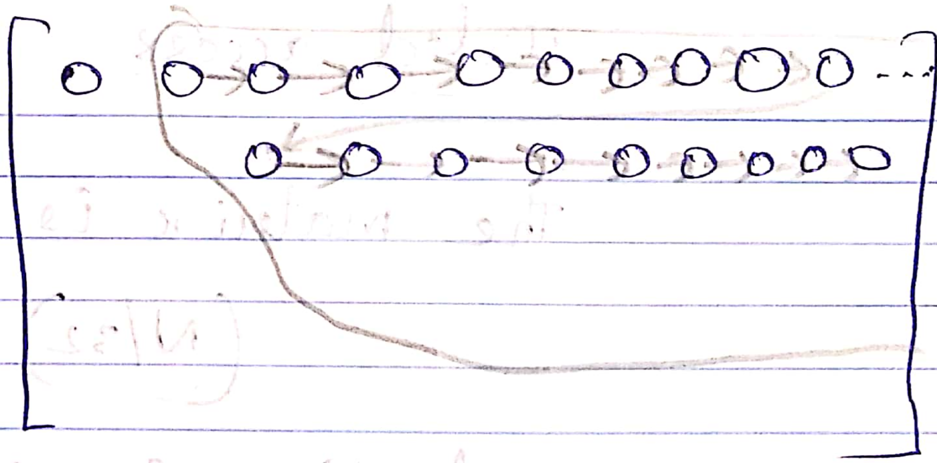
$$A[1][2] = \begin{bmatrix} 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 \\ 1 & 1 & 1 & 0 \end{bmatrix}$$

⋮

and so on

Tiled access

A [1] [1] =



A[1][2] . . . . . . A[1][9]

A[2][3] . . . . . . A[2][10]

A[3][4] . . . . . . A[3][11]

⋮ ⋮ ⋮

A[8][9] ⋮ A[8][17]

lu

$$\begin{bmatrix} A_{11} & A_{12} & A_{13} \\ A_{21} & A_{22} & A_{23} \\ A_{31} & A_{32} & A_{33} \end{bmatrix} = \begin{bmatrix} u_{11} & u_{12} & u_{13} \\ 0 & u_{22} & u_{23} \\ 0 & 0 & u_{33} \end{bmatrix} \begin{bmatrix} l_{11} & 0 & 0 \\ l_{21} & l_{22} & 0 \\ l_{31} & l_{32} & l_{33} \end{bmatrix}$$

in our algo,

$k = 0$

$j = 1$

$i = 1$

$\Rightarrow$ a[0][1] = a[0][1] / a[0][0]

a[1][1] = a[1][1] − (a[1][0] x a[0][1])

Tiled access

The matrix is split into (N/32) small matrices and the same operation is performed.

where lbv is the lower triangle vertices & ubv is upper triangle verticles.

Smaller access sizes than normal access as (N/32) small matrices are traversed to form the actual matrix