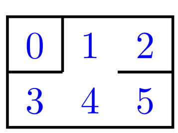
Boolean Matrices

- A *boolean matrix* is a 2-dimensional array of boolean values.
- Square boolean matrices are often used to store adjacency information.
- For example, we can number the cells in a maze from 0 to $n-1$, and define an $n \times n$ boolean matrix A such that $A[i,j]$ is **true** when cells i and j are adjacent with no intervening wall.

Example



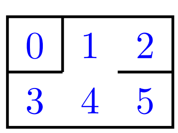
	0	1	2	3	4	5
0						
1			T		T	
2		T				
3					T	
4		T		T		T
5					T	

Empty matrix locations are **false**.

Reflexive Closure

- Given an adjacency matrix, suppose we set the elements on the main diagonal (i.e., all locations $[i, i]$) to **true**.
- This is known as the *reflexive closure*.
- A location $[i, j]$ is **true** when we can get from i to j in *at most* one step.
  - We can get from i to i in 0 steps.
  - We can get from i to j in exactly 1 step if i is adjacent to j.

Example



	0	1	2	3	4	5
0	T					
1		T	T		T	
2		T	T			
3				T	T	
4		T		T	T	T
5					T	T

Empty matrix locations are **false**.

Dot Product

- A *boolean vector* is a 1-dimensional array of boolean values.
  - **Example:** a row or column of a boolean matrix.
- The *dot product* of two boolean vectors of the same length is a boolean value that is **true** when there is some index i such that element i is **true** in both vectors.
  - $(T\ F\ T\ F) \cdot (F\ F\ T\ F) = T$
  - $(T\ F\ F\ T) \cdot (F\ T\ F\ F) = F$
- This is the same as the dot product for numeric vectors, only we use `||` instead of $+$ and `&&` instead of $\times$.

Application of Dot Product

- Suppose row i of a boolean matrix indicates which cells we can reach in at most m steps from cell i, and column j indicates the cells from which we can reach cell j in at most m steps.
- The dot product of row i and column j indicates whether we can reach cell j from cell i in at most $2m$ steps.
  - The dot product is **true** whenever there is a cell k such that we can reach k from i in at most m steps and we can reach j from k in at most m steps.

Squaring an $n \times n$ Boolean Matrix

- Given an $n \times n$ boolean matrix A, A^2 is the boolean matrix whose element $[i, j]$ is the dot product of row i and column j of A.

$$\begin{pmatrix} T & & & & & \\ & T & T & & T & \\ & T & T & & & \\ & & & T & T & \\ & T & & T & T & T \\ & & & & T & T \end{pmatrix}^2 = \begin{pmatrix} T & & & & & \\ & T & T & T & T & T \\ & T & T & & T & \\ & T & & T & T & T \\ & T & T & T & T & T \\ & T & & T & T & T \end{pmatrix}$$

Reflexive Transitive Closure

- If A is an $n \times n$ reflexive closure, then A^2 indicates when there is a path of length at most 2 from i to j.
- A^4 indicates when there is a path of length at most 4.
- A^k, when $k \geq n - 1$, indicates when there is any path from i to j.
- This matrix is called the *reflexive transitive closure.*

Implementation of Boolean Matrices

- We represent a boolean matrix as a singly-dimensioned array of boolean vectors.
- We represent a boolean vector as a linked list of integers.
- The integers in the list are the vector locations containing **true**.
- We maintain the list in decreasing order and terminate it with a cell containing -1 (in order to assist searching).
- In order to make inserting simpler, we begin each list with a *header cell* — a cell whose **Data** we don't use.

Example

To represent the row $(F\ T\ F\ T\ T\ T)$: