

# Minimal Spanning Tree Algorithm: In-the-matrix

## The Problem

Find the set of edges of minimum weight that spans the graph. To span a graph means to connect all the nodes in the graph.

## Algorithm

The MST algorithm builds the tree up by starting with a single node and then adding one edge to it on each iteration. It doesn't matter which node you begin with. The key is to add the edge to the closest unconnected node to the tree on each iteration.

| Algorithm step                                                                | Book-keeping in the matrix                                                      |
|-------------------------------------------------------------------------------|---------------------------------------------------------------------------------|
| 1. Begin by selecting any node arbitrarily and use it to initialize the tree. | Circle the node's row label.<br>Cross out the node's column.                    |
| 2. Identify the lightest edge that connects to a new (unconnected) node,...   | Look for the smallest number in the circled row ...                             |
| 3. ...and add this edge to the tree.                                          | ...Circle the smallest entry (let's say it is in column $i$ )                   |
| Continue until all nodes have been added.                                     | Circle the label for row $i$ .<br>Cross out the rest of entries in column $i$ . |

## Example

|   | A | B | C | D | E |
|---|---|---|---|---|---|
| A | — | 7 | 4 | — | — |
| B | 7 | — | 2 | 8 | — |
| C | 4 | 2 | — | 5 | 6 |
| D | — | 8 | 5 | — | 1 |
| E | — | — | 6 | 1 | — |

We begin by arbitrarily selecting A as our starting node so we “circle” its row label, and cross out its column (since we have “reached” it we won't ever connect to it again, because to do so would create a circuit).

|   | A                                                             | B            | C | D | E |
|---|---------------------------------------------------------------|--------------|---|---|---|
| A | <span style="border: 1px solid green; padding: 2px;">A</span> | <del>/</del> | 7 | 4 | — |
| B | <del>/</del>                                                  | —            | 2 | 8 | — |
| C | <del>/</del>                                                  | 2            | — | 5 | 6 |
| D | <del>/</del>                                                  | 8            | 5 | — | 1 |
| E | <del>/</del>                                                  | —            | 6 | 1 | — |

Now we can consider edges from A to unreached nodes. These are shown in green, 7 and 4, and we want the lightest one so we select the 4.

This means we will:

- Circle that element,
- Cross out the rest of the column, and
- Circle the corresponding row label, i.e. C.

|   | A            | B | C            | D | E |
|---|--------------|---|--------------|---|---|
| A | <del>/</del> | 7 | 4            | - | - |
| B | <del>/</del> | - | <del>/</del> | 8 | - |
| C | <del>/</del> | 2 | <del>/</del> | 5 | 6 |
| D | <del>/</del> | 8 | <del>/</del> | - | 1 |
| E | <del>/</del> | - | <del>/</del> | 1 | - |

Now we can consider edges from the nodes we have reached (A and C), i.e. we look in the rows with circled labels for the smallest entry: 2 in this case.

And we:

- Circle that element,
- Cross out the rest of the column, and
- Circle the corresponding row label, i.e. B.

|   | A            | B            | C            | D | E |
|---|--------------|--------------|--------------|---|---|
| A | <del>/</del> | <del>/</del> | 4            | - | - |
| B | <del>/</del> | <del>/</del> | <del>/</del> | 8 | - |
| C | <del>/</del> | 2            | <del>/</del> | 5 | 6 |
| D | <del>/</del> | <del>/</del> | <del>/</del> | - | 1 |
| E | <del>/</del> | <del>/</del> | <del>/</del> | 1 | - |

Now we look in the rows with circled labels (A, B, C) for the smallest entry: 5.

And we:

- Circle that element,
- Cross out the rest of the column, and
- Circle the corresponding row label, i.e. D.

|   | A            | B            | C            | D            | E |
|---|--------------|--------------|--------------|--------------|---|
| A | <del>/</del> | <del>/</del> | 4            | <del>/</del> | - |
| B | <del>/</del> | <del>/</del> | <del>/</del> | <del>/</del> | - |
| C | <del>/</del> | 2            | <del>/</del> | 5            | 6 |
| D | <del>/</del> | <del>/</del> | <del>/</del> | <del>/</del> | 1 |
| E | <del>/</del> | <del>/</del> | <del>/</del> | <del>/</del> | - |

Now we look in the rows with circled labels (A, B, C, D) for the smallest entry: 1.

And we:

- Circle that element,
- Cross out the rest of the column, and
- Circle the corresponding row label, i.e. E.

|   | A            | B            | C            | D            | E            |
|---|--------------|--------------|--------------|--------------|--------------|
| A | <del>/</del> | <del>/</del> | 4            | <del>/</del> | <del>/</del> |
| B | <del>/</del> | <del>/</del> | <del>/</del> | <del>/</del> | <del>/</del> |
| C | <del>/</del> | 2            | <del>/</del> | 5            | <del>/</del> |
| D | <del>/</del> | <del>/</del> | <del>/</del> | <del>/</del> | 1            |
| E | <del>/</del> | <del>/</del> | <del>/</del> | <del>/</del> | <del>/</del> |

Since all the row labels are circled we have reached/connected all the vertices in the graph.

The MST consists of the edges connecting, A-C @ 4, C-B @ 2, C-D @ 5, and D-E @ 1, for a total weight of 4+2+5+1=12.