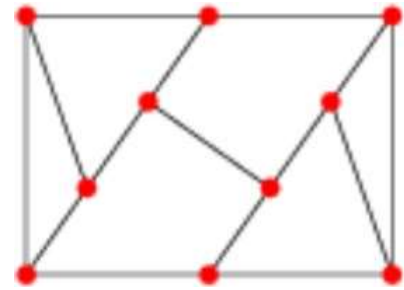


Project: Grapher

The aim of the project is to develop a graph application.



General Information

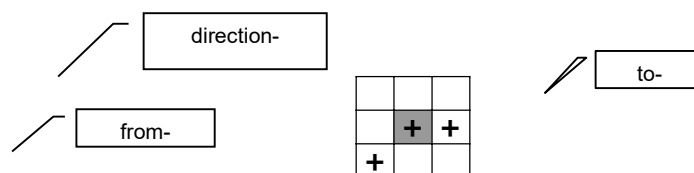
The application has a board with 25*40 squares. In this application; user can draw, load, save a graph, and calculate R , R^2 , R^3 , ..., R^n , R^* and R_{\min} matrices. R matrix gives directly connected nodes (1 step away). R^2 matrix gives exactly 2 steps away points. R^n matrix gives exactly n steps away points. R^* matrix gives all connected points. R_{\min} matrix gives the minimum number of steps required for going point a to point b .

Graph Drawing and Matrix Calculation Stages

1. Graph can be drawn manually or loaded from "graph.txt" file.
Loaded graph is shown on the screen, additional drawing is possible. A graph can also be saved.
2. R relation matrix (adjacency matrix) is formed by tracing the graph.
3. Calculate R^2 , R^3 , ..., R^n matrices for the graph (n : the number of nodes/points)
4. Calculate R^* (transitivity closure or connectivity relation) matrix.
5. Calculate R_{\min} matrix which shows the minimum number of steps required for going point a to point b .
6. Save R , R^2 , R^3 , ..., R^n , R^* and R_{\min} matrices to the "matrix.txt" file.

Graph Drawing Rules

- Graph board has 25*40 squares.
- The first 16 letters of the alphabet (A, B, C, D, E, F, G, H, I, J, K, L, M, N, O, P) can represent nodes of the graph. User must use the first n letters for an n -node graph.
- Each node can use 8 neighbor squares (horizontal/vertical/diagonal) to connect to other nodes.
These 8 squares are only used for going into the node or going out of the node, not for any other line transit passings.
- Edges/lines are represented by $+$ and \mathbf{X} symbols. Going into a node is represented by \mathbf{X} symbol.
- Edges go straight in 8 directions (horizontal/vertical/diagonal) or can change direction.
Direction-change-square must have max 2 out of 8 neighbor squares with $+$ or \mathbf{X} symbols, one for from-square, one for to-square.



Keyboard Commands

Drawing Keys:

- **Cursor movement keys:** To move the cursor on the graph
- **Letters A-P:** Put a node to the cursor's position
- **X :** Put an ending edge part (\mathbf{X}) to the cursor's position
- **space :** Put an edge part ($+$) to the cursor's position
- **.** : Delete the symbol at the cursor's position (. symbol = empty square)

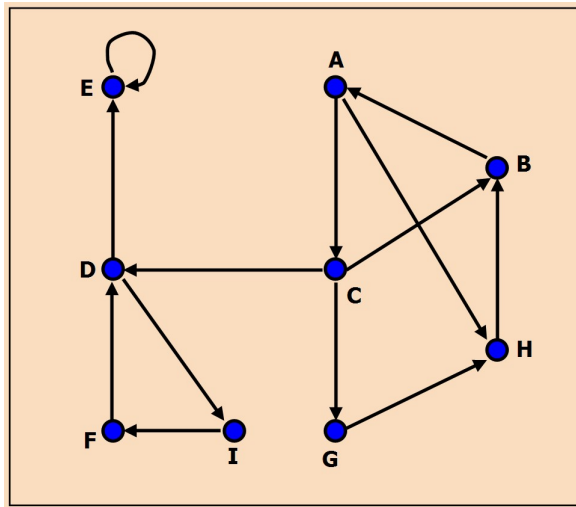
Calculation and Display Keys:

- **1 :** Trace the graph to form R relation matrix.
Calculate R^2 , R^3 , ..., R^n , R^* and R_{\min} matrices.
Show R matrix on the top right section of the screen.
Show R^* matrix on the bottom right section of the screen.
- **2-9 :** Show R^2 , R^3 , ... or R^n matrix on the bottom right section of the screen.
- **0 :** Show R_{\min} matrix on the bottom right section of the screen.

Query key:

- **Q** : Takes 2 nodes (a and b), and returns min number of steps required for going node a to node b.

Sample Graph



Sample Screen of the Application

Screen Arrangement

- Left : 25*40-square Graph Board
- Top right: R relation matrix
- Bottom right: $R^2, R^3, \dots, R^n, R^*$ or R_{\min} matrix (default R^*)

```

1234567890123456789012345678901234567890
#####
1#.....#
2#.....#
3#...+++.....#
4#...+.+......#
5#...EX+.....AX+++++++.....#
6#...X.....++.+......#
7#...+.+.+.+......#
8#...+.+.+.+.B.....#
9#...+.+.+.+.XX.....#
0#...+.+.+.+.+.+......#
1#...+.X.....+.+......#
2#...DX+++++++C+++++.+......#
3#...X+.....+.+.+......#
4#...+.+.+.+.+.+......#
5#...+.++++.+.+.+......#
6#...+.+.+.+.++++XH.....#
7#...+.+.+.+.X.....#
8#...+.X.....X.....#
9#...FX++++I.....G+++++++.....#
0#.....#
1#.....#
2#.....#
3#.....#
4#.....#
5#.....#
#####

R matrix
-----
A B C D E F G H I
A 0 0 1 0 0 0 0 0 1 0
B 1 0 0 0 0 0 0 0 0
C 0 1 0 1 0 0 1 0 0
D 0 0 0 0 1 0 0 0 1
E 0 0 0 0 1 0 0 0 0
F 0 0 0 1 0 0 0 0 0
G 0 0 0 0 0 0 0 1 0
H 0 1 0 0 0 0 0 0 0
I 0 0 0 0 0 1 0 0 0

R* matrix
-----
A B C D E F G H I
A 1 1 1 1 1 1 1 1 1
B 1 1 1 1 1 1 1 1 1
C 1 1 1 1 1 1 1 1 1
D 0 0 0 1 1 1 0 0 1
E 0 0 0 0 1 0 0 0 0
F 0 0 0 1 1 1 0 0 1
G 1 1 1 1 1 1 1 1 1
H 1 1 1 1 1 1 1 1 1
I 0 0 0 1 1 1 0 0 1

Query for min steps:  from: F      to: H

Answer: Not possible

```