



# *PROJECT – III*

## *GRAPHER*

CME1251 PROJECT BASED LEARNING-I

BY

2018510016 SADULLAH CİHAN

2018510023 ÖZCAN ELMACI

2018510013 MEHMET FATİH BİRDİR



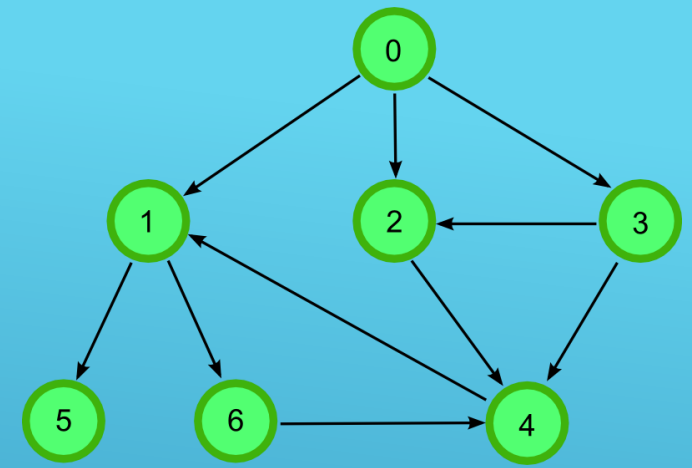
# OUTLINE

- ▶ INTRODUCTION
- ▶ PROGRESS SUMMARY
  - ▶ Requirements
  - ▶ Task Sharing, Scheduling
  - ▶ Completed Tasks
  - ▶ Incomplete Tasks: Reasons, Explanations
  - ▶ Additional Improvements
  - ▶ Problems Encountered
  - ▶ Algorithms and Solution Strategies
  - ▶ Screenshots
  - ▶ Conclusion, References



# INTRODUCTION

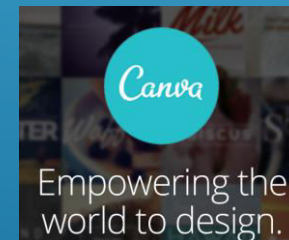
- ▶ *25\*40 board*
- ▶ *Just 'directed graph'*
- ▶ *User can draw, load, save a graph*
- ▶ *Also  $R$ ,  $R^2$ ,  $R^3$  ...  $R_n$ ,  $R^*$ ,  $R$  min*



# PROGRESS SUMMARY

## REQUIREMENTS

- ▶ *C# knowledge*
- ▶ *Visual Studio*
- ▶ *Teamwork*
- ▶ *Trello*
- ▶ *Canva*



# PROGRESS SUMMARY

## Task Sharing

*Sadullah Cihan :*

- ▶ Loading
- ▶ Redrawing
- ▶ Tracing
- ▶ Forming R matrix
- ▶ Calculating RN
- ▶  $R^*$  and Rmin
- ▶ Matrix saving
- ▶ PowerPoint
- ▶ Reports

*Özcan Elmacı :*

- ▶ Drawing
- ▶ Tracing
- ▶ Forming R matrix
- ▶ Calculating RN
- ▶ Saving the graph
- ▶  $R^*$  and Rmin
- ▶ Matrix saving
- ▶ Reports
- ▶ Testing

*M.Fatih Birdir:*

- ▶ Necessary variables
- ▶ Tracing
- ▶ Testing



# PROGRESS SUMMARY

## Scheduling

### First Week

- ▶ Meeting with project partners.
- ▶ Analyzing the project and discussing solution alternatives.
- ▶ Starting to code with creating necessary variables/structures.

### Second Week

- ▶ Drawing the graph.
- ▶ Loading the graph.



# PROGRESS SUMMARY

## Scheduling

### ► Third Week

- Tracing the graph
- Forming  $R$  matrix

### ► Fourth Week

- Calculating  $R, R^2, R^3 \dots R_n$  (RECURSIVE)
- Saving the graph

### ► Fifth Week

- Calculating  $R^*$ ,  $R$  min and saving matrices.
- Designing a poster.
- Preparing for the presentation.





# PROGRESS SUMMARY

## Completed Tasks

- ▶ Creating necessary variables.
- ▶ Draving
- ▶ Loading
- ▶ Tracing and forming  $R$  matrix
- ▶  **$R\_N()$  : Recursive** matrix multiplication
- ▶ Saving the graph
- ▶ Calculating  $R^*$  and  $R_{min}$
- ▶ Saving matrices





# PROGRESS SUMMARY

## *Incomplete Tasks*

- ▶ *There is no incomplete task.*
- ▶ *We have finished the project completely.*



# PROGRESS SUMMARY

## *Additional Improvements*



► *Font color*

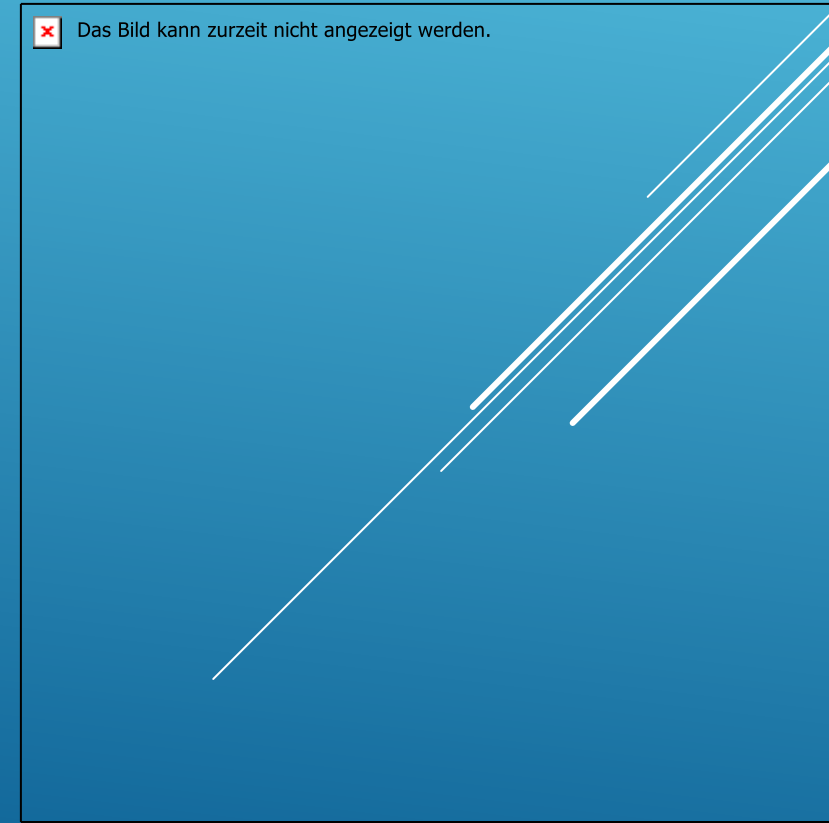
```
Console.ForegroundColor = ConsoleColor.Blue;
```

► *Recursive matrix multiplication*



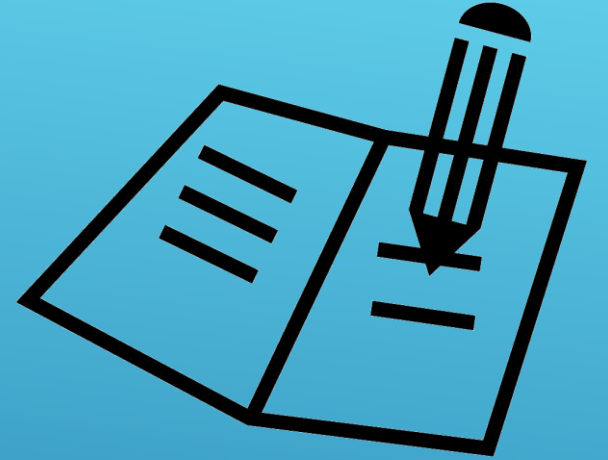
# PROBLEMS ENCOUNTERED

- ▶ Insufficient knowledge 'File Operations'
- ▶ Designing algorithm (tracing)
- ▶ Hidden ASCII characters
- ▶ 'Console.SetCursorPosition' X Array
- ▶ Leck of time...



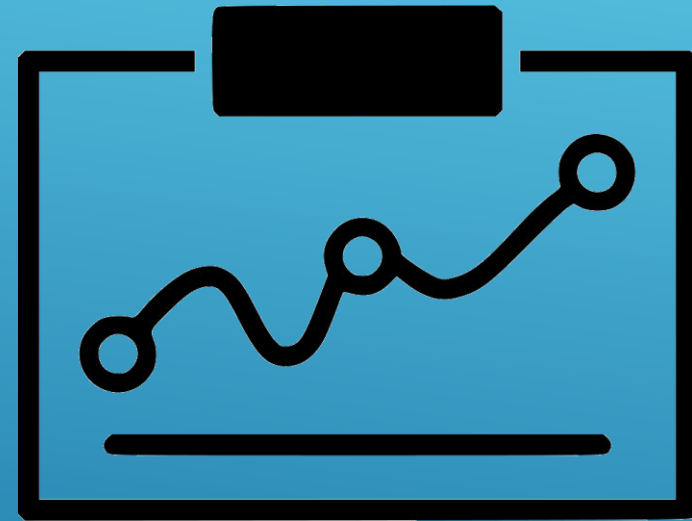
# ALGORITHM AND SOLUTION STRATEGIES

- ▶ Efficient
- ▶ Maintainable
- ▶ Recursive

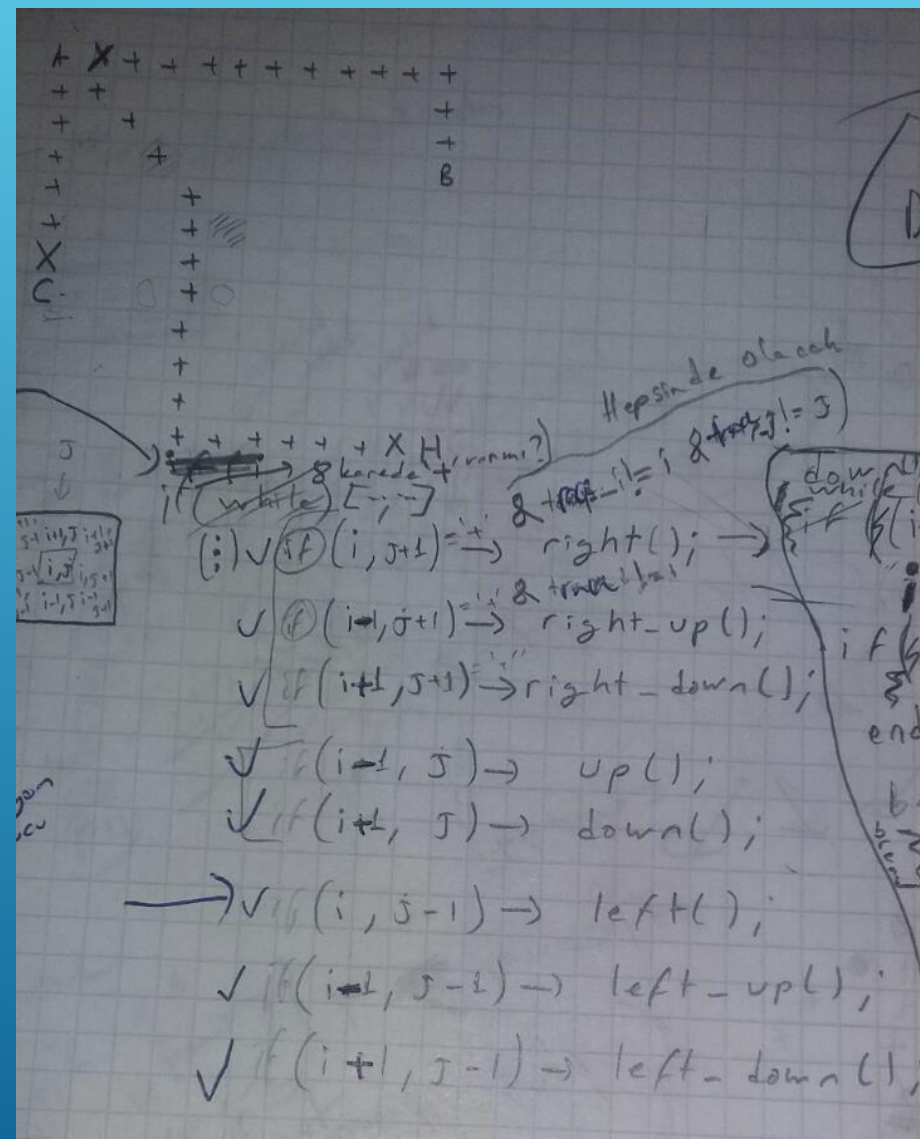
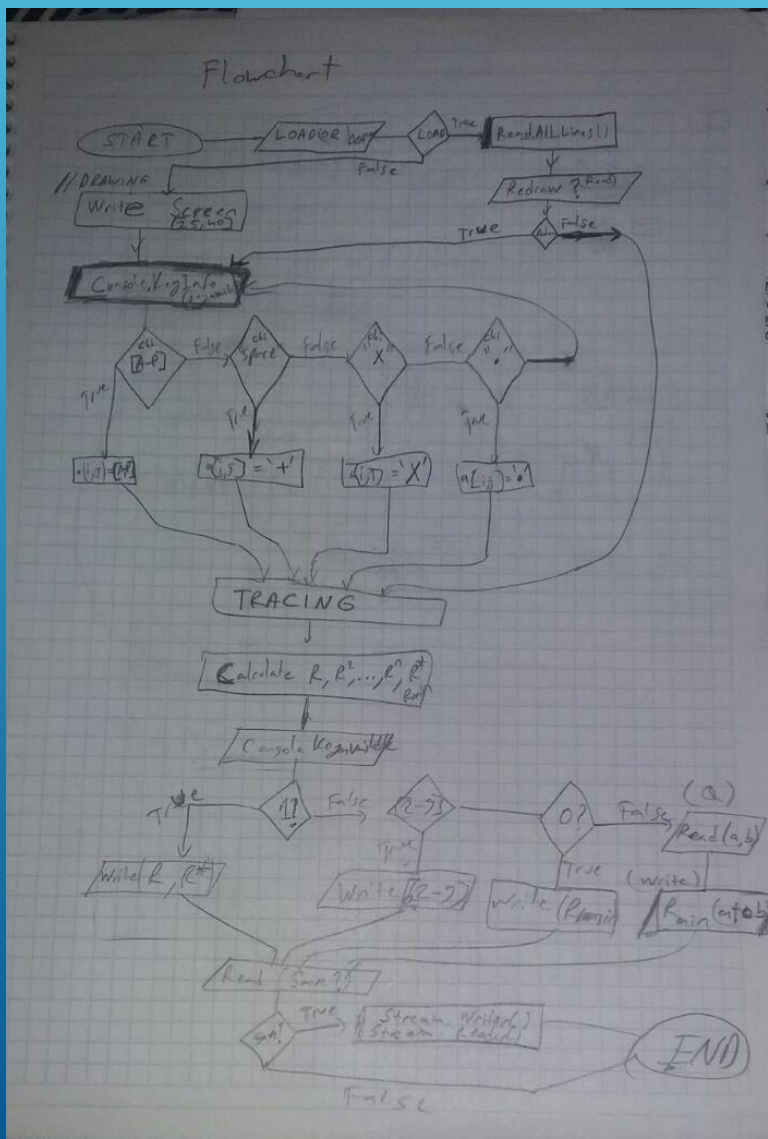


# ALGORITHM AND SOLUTION STRATEGIES

```
.....
...+++.....
...+.+.....
...EX+.....AX+++++.....
...X.....++.....+.
...+.....+.+.....+.
...+.....+.+.....B.
...+.....+.+.....XX.
...+.....+.+.....+.+.
...+.....X.....+.+.
...DX++++++C+++++.....+.
...X+.....+.+.+.
...+.+.....+.+.+.
...+.++++.....+.+.+.
...+.+.+.+.+++++XH.
...+.+.+.+.+.X.
...+.X.....X.....+.
...FX++++I.....G++++++.....
```

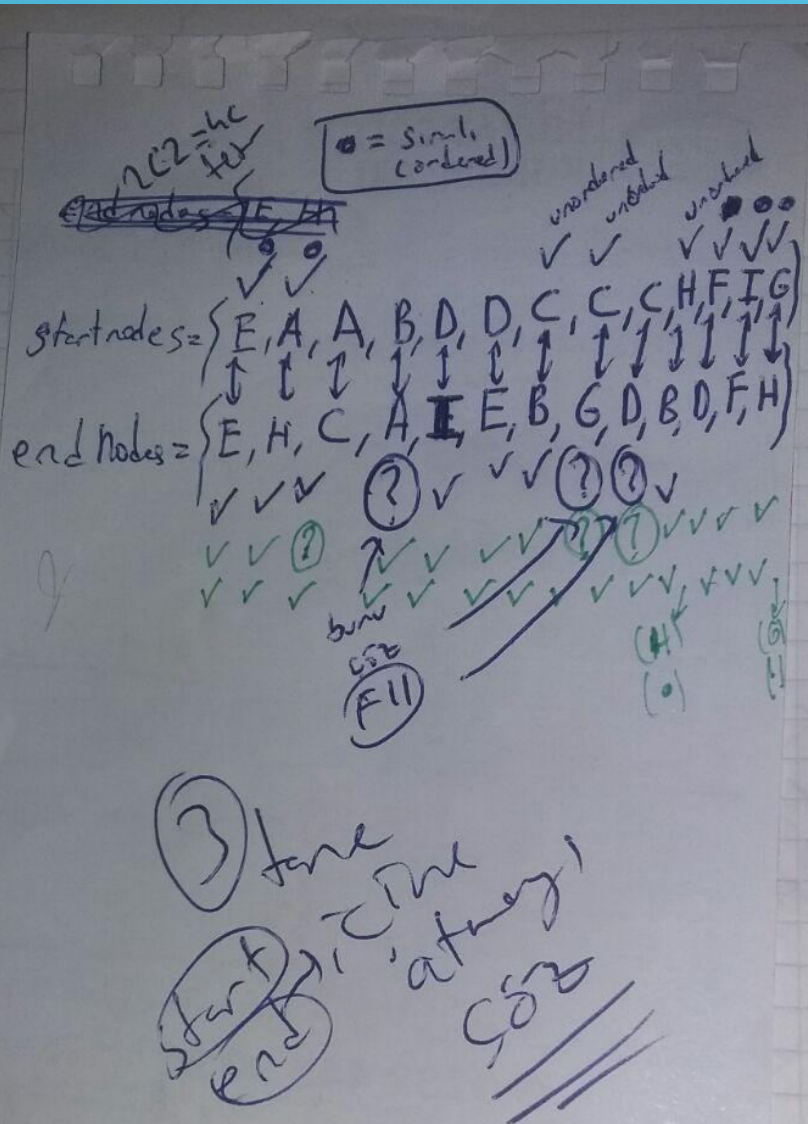


# ALGORITHM AND SOLUTION STRATEGIES





# ALGORITHM AND SOLUTION STRATEGIES



Handwritten notes on a piece of paper:

$R(1) \Rightarrow R - \text{Matrix}$

Recursive Algo.

state int C, J       $R_n(\text{int } N)$

```

{
  if (n == 1) return R(1);
  else
  {
    Multiplication(inside)
    :
     $R_n(N-1)[i, k] * R(1)[k, j];$ 
    :
  }
}

```

$R2 = R(2-1) - R(1)$

$R3 = R(2) - R(1)$

# SCREENSHOTS

## ► Main manu

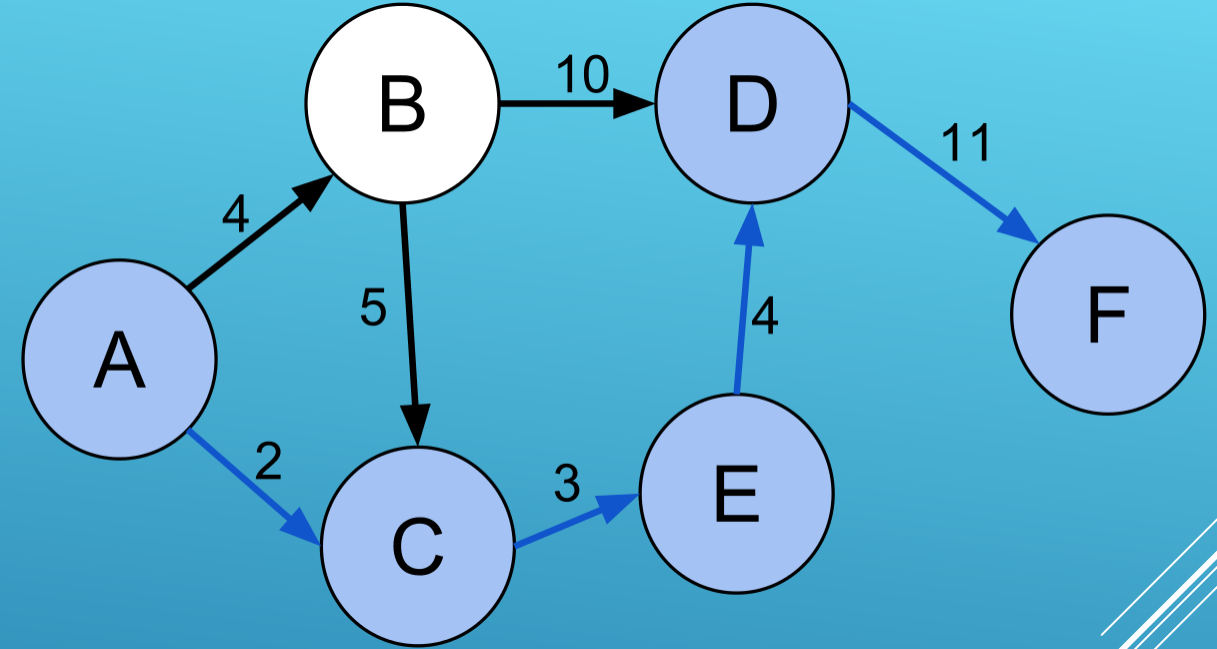
```
*****THE GRAPHER*****
```

```
-MENU-
```

```
1 - LOAD GRAPH  
2 - DRAW GRAPH MANUALLY
```

```
NOTE: YOU SHOULD USE UPPERCASE LETTER  
WHILE DRAWING OR REDRAWING THE GRAPH
```

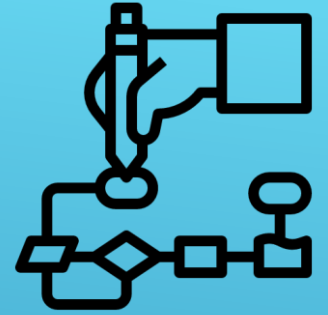
```
PLEASE CHOOSE MODE:
```



```
Enter the option of which graph you want to see  
1-Existed graph  
2-New existed graph
```

# SCREENSHOTS

► Inside menu

[illegible]

**MENU**

- '0' => Show R Min Matrix
- '1' => Show R and R \* Matrix
- '2-9' => Show Rn Matrix
- 'Q' => Query For Min Steps
- 'C' => Change The Graph

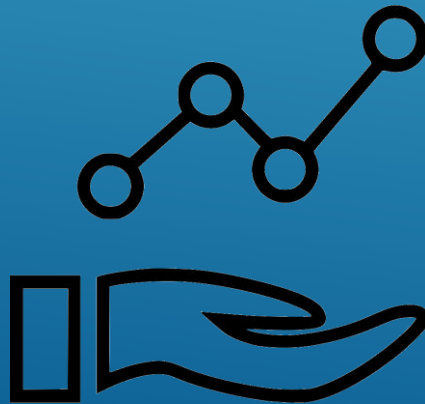
Saving operation has finished successfully !



# SCREENSHOTS

## ► Query and Rmin

```
QUERY FOR MIN STEPS
From: E To: E
1
```



R MATRIX		R MIN	
	ABCDEFGHIJKLMN		ABCDEFGHIJKLMN
A	0010000100000000	A	3212342130000000
B	1000000000000000	B	1323453240000000
C	0101001000000000	C	2131231220000000
D	0000100010000000	D	0003120010000000
E	0000100000000000	E	0000100000000000
F	0001000000000000	F	0001230020000000
G	0000000100000000	G	3245675160000000
H	0100000000000000	H	2134564350000000
I	0000010000000000	I	0002310030000000
J	0000000000000000	J	0000000000000000
K	0000000000000000	K	0000000000000000
L	0000000000000000	L	0000000000000000
M	0000000000000000	M	0000000000000000
N	0000000000000000	N	0000000000000000
O	0000000000000000	O	0000000000000000
P	0000000000000000	P	0000000000000000

### MENU

```
'0' => Show R Min Matrix
'1' => Show R and R * Matrix
'2-9' => Show Rn Matrix
'Q' => Query For Min Steps
'C' => Change The Graph
0
```

## ► SCREENSHOTS

► Saving

```
#####
#.....#
#.....#
#.....+++.....#
#.....+.+.....#
#.....EX+.....AX+++++++.....#
#.....X.....++.....+.....#
#.....+.....+.+.....+.....#
#.....+.....+.+.....B.....#
#.....+.....+.+.....XX.....#
#.....+.....+.+.....+.+.....#
#.....+.....X.....+.+.....+.....#
#.....DX+++++++C+++++++.....+.....#
#.....X+.....+.+.....+.....#
#.....+.+.....+.+.....+.....#
#.....+.++++.....+.+.....+.....#
#.....+.+.....+.+.....++++++XH.....#
#.....+.+.....+.+.....X.....#
#.....+.X.....X.....+......#
#.....FX++++I.....G+++++++.....#
#......#####
#.....#
#.....#
#.....#
#.....#
#####
press enter to save new graph
```

```
#####
# .....#
# .....#
# ...+++.....#
# ...+.+. ....#
# ...EX+.....AX+++++++.....#
# ...X.....++.....+.....#
# ...+.....+.+. ....+.....#
# ...+.....+.+. ....B.....#
# ...+.....+.+. ....XX.....#
# ...+.....+.+. ....+.+. ....#
# ...+.....X.....+.+. ....#
# ...DX+++++++C+++++++.....+.....#
# ...X+.....+.+. ....+.....#
# ...+.+. ....+.+. ....+.....#
# ...+.+++++.....+.+. ....+.....#
# ...+.+. ....+.++++++XH.....#
# ...+.+. ....+. ....X.....#
# ...+. ....X.....X.....+. ....#
# ...FX++++I.....G+++++++.....#
# .....+. ....#
# .....+. ....#
# .....+. ....#
# .....X.....#
# .....p.....#
# .....#
#####
Saving operation has finished successfully !
press enter to save new graph
```

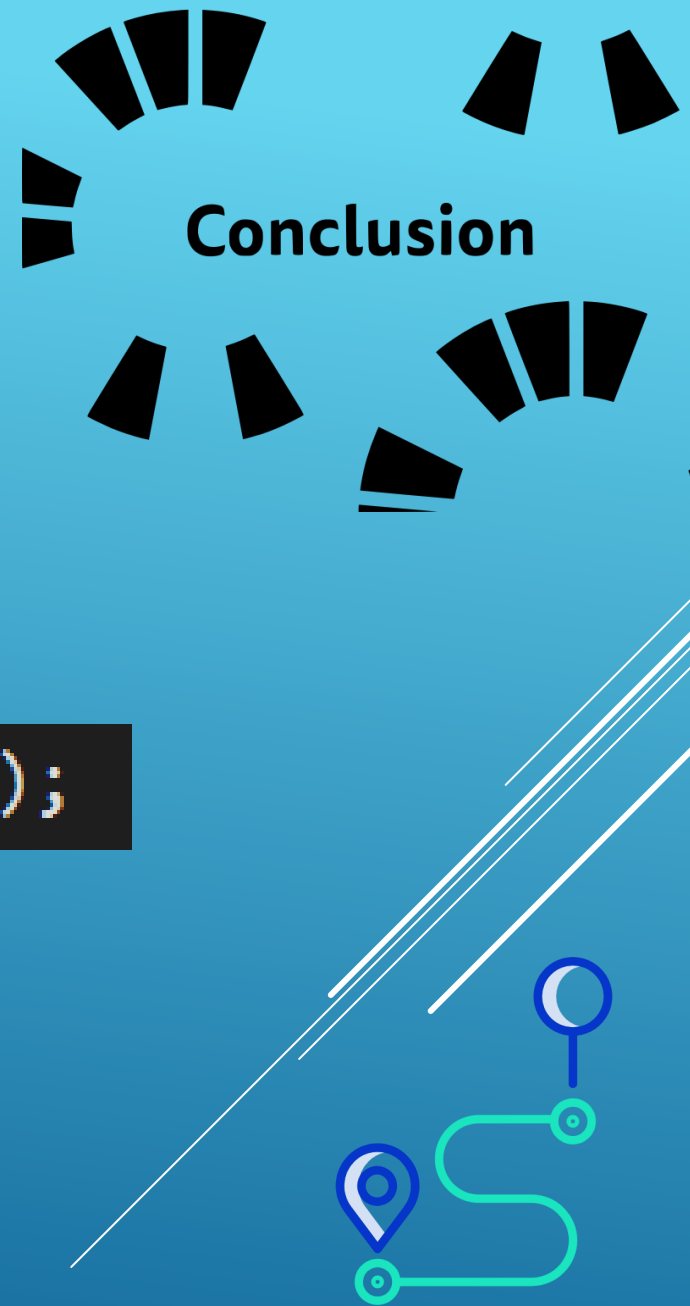
# CONCLUSION

*We have learned:*

- ▶ Usage of “Functions, Procedures, and Recursive”
- ▶ Team work (Like IT Company)
- ▶ New functions such as,

```
StreamReader graph_txt = File.OpenText(path);
```

- ▶ *The project has been successfully finished on time exactly*





# REFERENCES

- ▶ <https://docs.microsoft.com/enus/dotnet/csharp/programming-guide/>
- ▶ <https://www.canva.com>



Lastly...



# RECURSIVE ALGORITHM



## GRAPHER

### INTRODUCTION

- The application has a board with 25\*40 squares.
- In this application; user can draw, load, save a graph, and calculate R, R2, R3, ..., Rn, R\* and Rmin matrices.
- R matrix gives directly connected nodes (1 step away).  
R2 matrix gives exactly 2 steps away points.
- Rn matrix gives exactly n steps away points. R\* matrix gives all connected points.
- Rmin matrix gives the minimum number of steps required for going point a to point b.

### WELL DESIGNED CONSOLE INTERFACE

```
R MATRIX      R*
ABCDEFGHIJKLMN ABCDEFGHIJKLMN
A 0010000100000000 A 1111111110000000
B 1000000000000000 B 1111111110000000
C 0101001000000000 C 1111111110000000
D 0000100010000000 D 0001110010000000
E 0000100000000000 E 0000100000000000
F 0001000000000000 F 0001110010000000
G 0000000100000000 G 1111111110000000
H 0100000000000000 H 1111111110000000
I 0000010000000000 I 0001110010000000
J 0000000000000000 J 0000000000000000
K 0000000000000000 K 0000000000000000
L 0000000000000000 L 0000000000000000
M 0000000000000000 M 0000000000000000
N 0000000000000000 N 0000000000000000
O 0000000000000000 O 0000000000000000
P 0000000000000000 P 0000000000000000

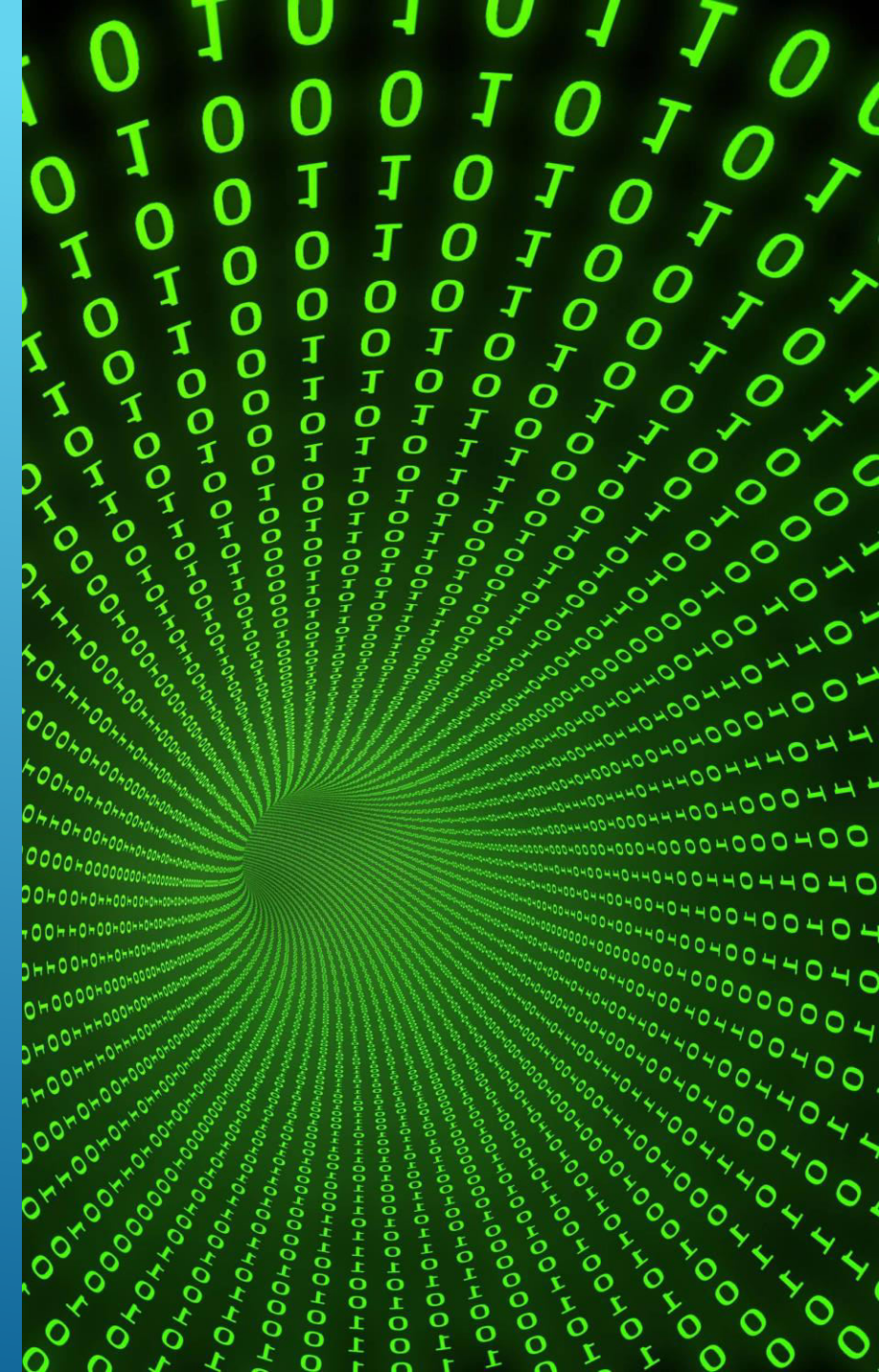
MENU          QUERY FOR MIN STEPS
'0' => Show R Min Matrix      From: E To: E
'1' => Show R and R * Matrix  1
'2-9' => Show Rn Matrix
'Q' => Query For Min Steps
'C' => Change The Graph
```

### RECURSIVE MATRIX MULTIPLICATION



2018510016 SADULLAH CİHAN  
2018510023 ÖZCAN ELMACI  
2018510013 M.FATİH BİRDİR

**DEUCENG**  
Dokuz Eylül University  
Dept. of Computer Engineering



THE END

