

PROJECT – III

SOLICENG

CME1252 Project Based Learning - II

BY

2020510010 ARDA ASLANER

2020510034 YUSUF GASSALOĞLU

2020510066 GÜNEY SÖĞÜT

2020510022 YUNUS EMRE DANGAÇ

Outline

INTRODUCTION

PROGRESS SUMMARY

- Requirements
- Task Sharing
- Scheduling
- Completed Tasks
- Additional Improvements

PROBLEMS ENCOUNTERED

ALGORITHMS AND SOLUTION STRATEGIES

SCREENSHOTS

CONCLUSION

QUESTIONS

REFERENCES

* A number stacking game

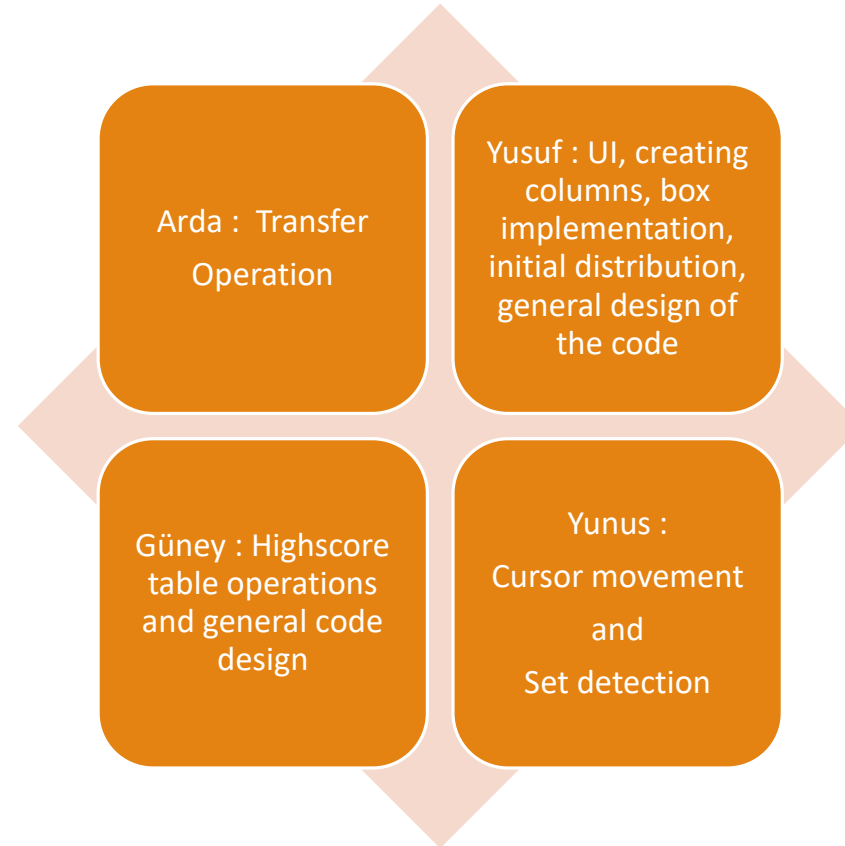
- [illegible]

Progress Summary: Requirements

- Transfer Algorithm
- Box Algorithm
- Enigma Java Library
- Linked Lists
(Single, Double & Multi)



Progress Summary: Task Sharing



Progress Summary : Scheduling

The tasks are completed on scheduled dates.

| Task \ Week | 1 | 2 | 3 | 4 | 5 |
|---|---|---|---|---|---|
| Design of classes, data structures, screen. Load operations. | X | | | | |
| Box implementation. Initial distribution of the box on the screen. | | X | | | |
| High Score Table | | | X | | |
| Columns. Transfer operations. | | | | X | |
| Remaining parts of the game, debugging/testing. | | | | | X |

Progress Summary : Completed Tasks

All tasks (Gameplay, score, etc.) were completed.



Progress
Summary:
Additional
Improvements



Well-designed UIX

Easy cursor movement

Sounds

Language selection

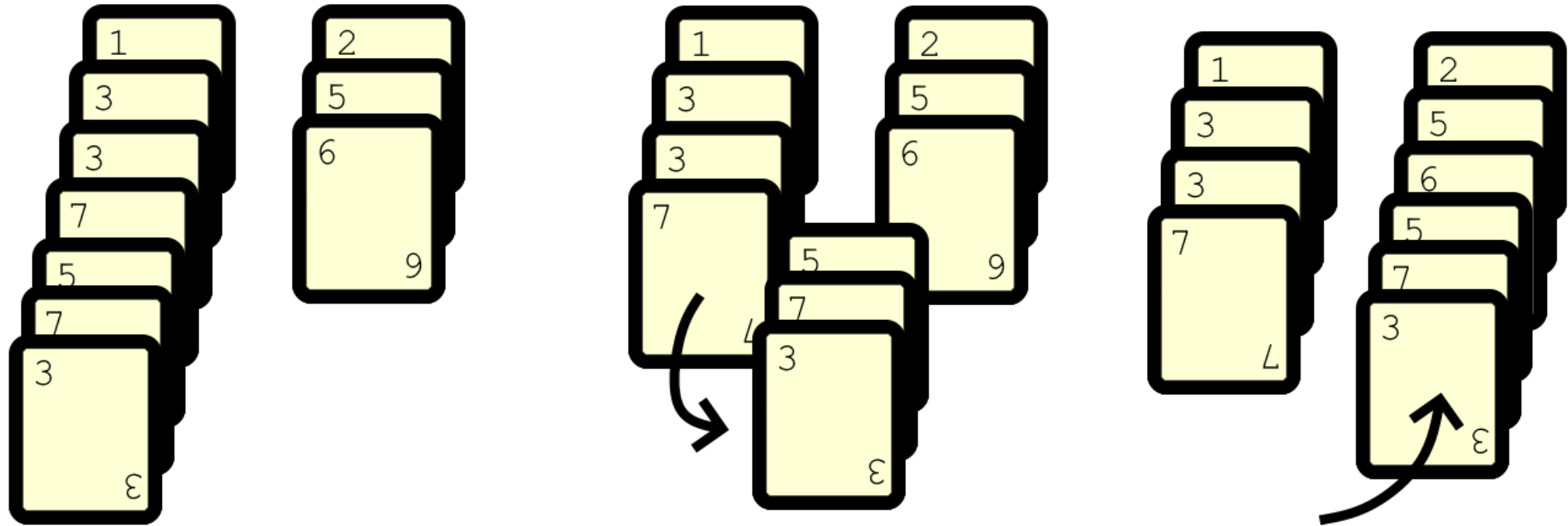


Problems Encountered

No Problems Encountered

Algorithms and Solution Strategies:

Transfer operation implementation



Algorithms and Solution Strategies:

Set finder

10 9 8 10 9 8 7 6 5 4 3 2 1 2 2



Found set

10 9 8 10 9 8 7 6 5 4 3 2 1

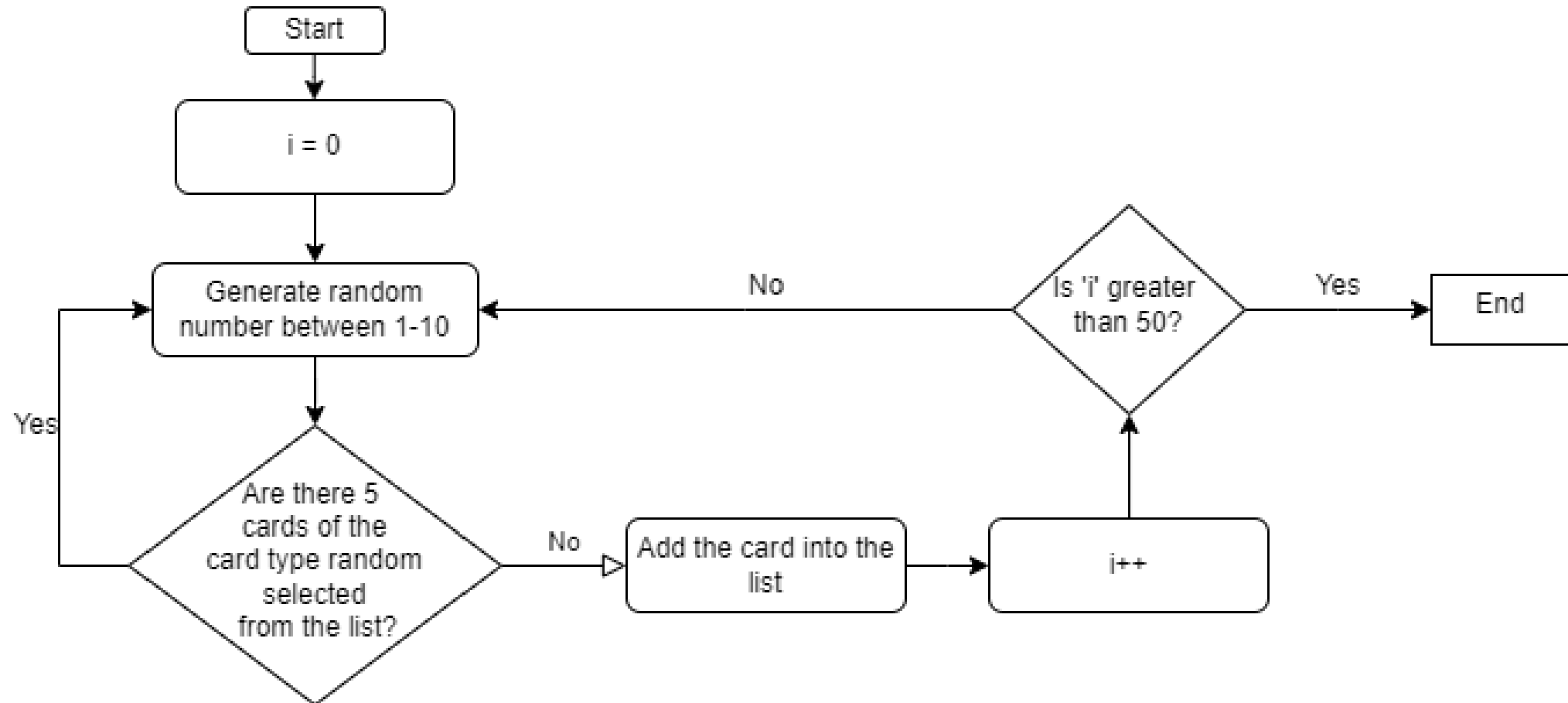
2 2



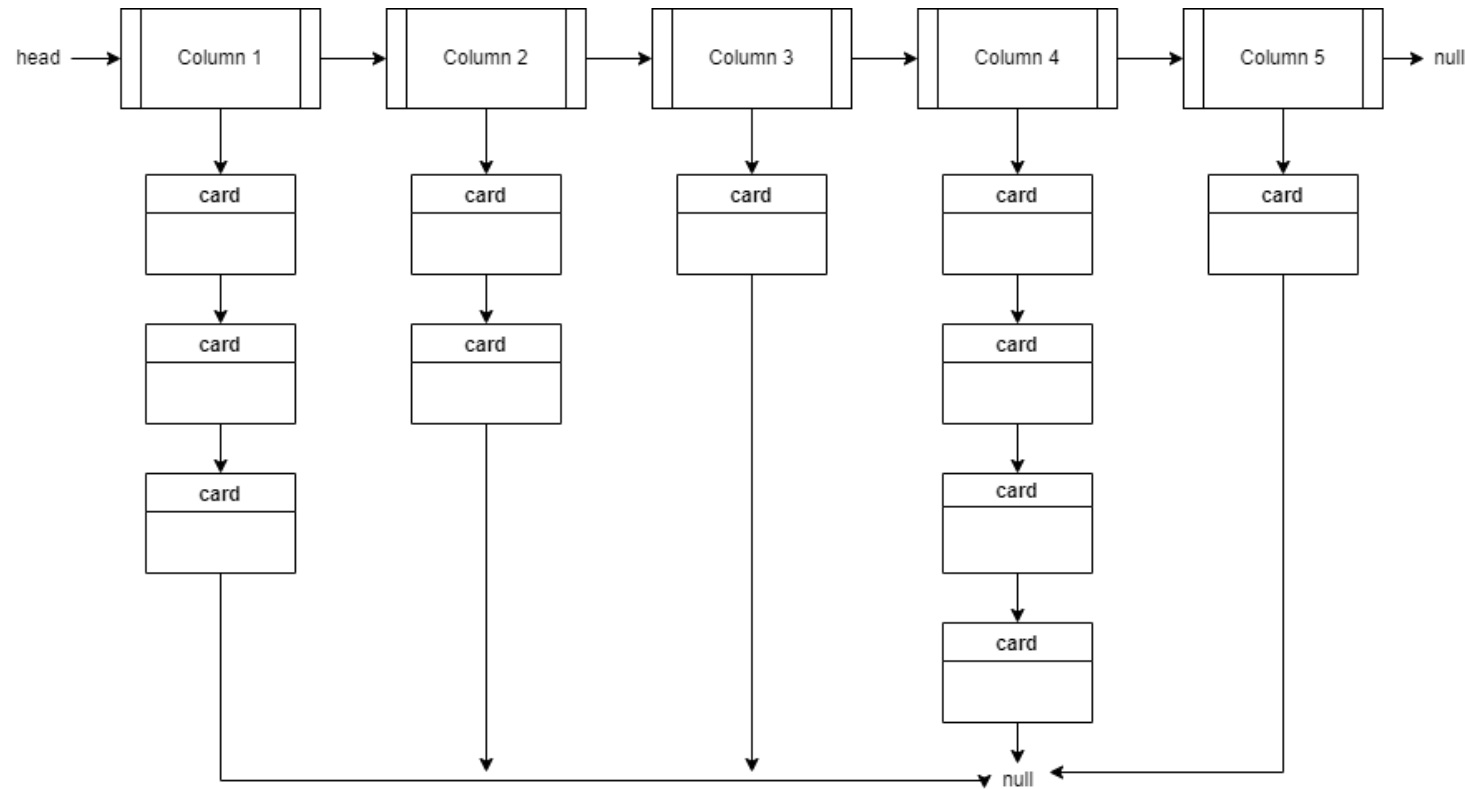
10 9 8 2 2

Algorithms and Solution Strategies:

Box Algorithm



Algorithms and Solution Strategies: Column Algorithm (Main Game)



Algorithms and Solution Strategies:

Sorting Algorithm

1500 2500 2000 1000



2500 1500 2000 1000



2500 2000 1500 1000

Screenshots

|7.---. |
| :/\: |
| () |
| *--*7|
`-----`

|9.---. |
| :/\: |
| () |
| *--*9|
`-----`

.*_____\ |.*_____\ |
| (_____) / ._____\ |
.*_____\ | |_____\ |
| _____) | _____\ |
_____. * _____. *
_____. * _____. *

_____|_____|
|_____|_____|
|_____|_____|
|_____|_____|
|_____|_____|
|_____|_____|

_____|_____|
|_____|_____|
|_____|_____|
|_____|_____|
|_____|_____|
|_____|_____|

.*_____\ |.*_____\ |
| (_____) / ._____\ |
.*_____\ | |_____\ |
| _____) | _____\ |
_____. * _____. *
_____. * _____. *

_____|_____|
|_____|_____|
|_____|_____|
|_____|_____|
|_____|_____|
|_____|_____|

_____|_____|
|_____|_____|
|_____|_____|
|_____|_____|
|_____|_____|
|_____|_____|

.*_____\ |.*_____\ |
| (_____) / ._____\ |
.*_____\ | |_____\ |
| _____) | _____\ |
_____. * _____. *
_____. * _____. *

|5.---. |
| :/\: |
| \ / |
| *--*7|
`-----`

|8.---. |
| :/\: |
| \ / |
| *--*9|
`-----`

_____|_____|_____|_____|
| . \ | | | | | | | | | |
| ____/ | | | | | | | | | |
| | | | | | | | | | | |

_____|_____|_____|_____|_____|_____|_____|_____|
| . | | . \ | | | | | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | | | | | | | | | | |
|_____* | | | | | | | | | | | | | | | | | | | | | | | | | | | |
|_____* | | | | | | | | | | | | | | | | | | | | | | | | | | | |

_____|_____|_____|_____|_____|_____|_____|_____|_____|_____|
| | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
|_____|_____|_____|_____|_____|_____|_____|_____|_____|_____|

_____|_____|_____|_____|_____|_____|_____|_____|_____|_____|
|_____|_____|_____|_____|_____|_____|_____|_____|_____|_____|
|_____|_____|_____|_____|_____|_____|_____|_____|_____|_____|
|_____|_____|_____|_____|_____|_____|_____|_____|_____|_____|



```
.-----.  
|5.--. |  
.-----.  
|4.--. |  
| :/\: |  
| ( ) |  
| '---'4|  
`-----`
```



```
.-----.  
|10.-. |  
.-----.  
|6.--. |  
.-----.  
|5.--. |  
.-----.  
|1.--. |  
.-----.  
|2.--. |  
.-----.  
|10.-. |  
| :/\: |  
| ( ) |  
| '---'10|  
`-----`
```



```
.-----.  
|1.--. |  
.-----.  
|3.--. |  
.-----.  
|10.-. |  
.-----.  
|8.--. |  
.-----.  
|8.--. |  
.-----.  
|1.--. |  
.-----.  
|2.--. |  
| :/\: |  
| ( ) |  
| '---'2|  
`-----`
```



```
.-----.  
|2.--. |  
.-----.  
|7.--. |  
.-----.  
|6.--. |  
.-----.  
|5.--. |  
.-----.  
|10.-. |  
.-----.  
|2.--. |  
.-----.  
|5.--. |  
| :/\: |  
| ( ) |  
| '---'5|  
`-----`
```



```
.-----.  
|5.--. |  
.-----.  
|9.--. |  
.-----.  
|2.--. |  
.-----.  
|1.--. |  
.-----.  
|1.--. |  
.-----.  
|4.--. |  
| :/\: |  
| ( ) |  
| '---'4|  
`-----`
```

press ESC to GIVE UP



Transfer : 4
Finished Sets : 0
Score : 0

Z: Select The Card
X: Release The Cards
C: Reset The Selection
V: Draw A Card

Ahmet 2520.0
Tarik Deniz 521.07
Sema Firat 378.24
Seda Ruzgar 224.67
Ali Vadi 136.32

Ahmet 2520.0
Tarik Deniz 521.07
Sema Firat 378.24
Seda Ruzgar 224.67
Ali Vadi 136.32

Your Score : 2520.0

Please press any letter then press enter to go to main menu.

Conclusion

- Management of stack of cards.
- Linked Lists.



Any Questions?

References

- ❖ <https://www.javatpoint.com/java-program-to-create-and-display-a-singly-linked-list#:~:text=The%20singly%20linked%20list%20is,next%20node%20in%20the%20list.>
- ❖ <https://www.geeksforgeeks.org/merge-sort-for-doubly-linked-list/>
- ❖ [https://en.wikipedia.org/wiki/UTF-8#:~:text=Defined%20by%20the%20Unicode%20Standard,Transformation%20Format%20%E2%80%93%208%2Dbit.&text=UTF%2D8%20is%20capable%20of,8%2Dbit\)%20code%20units](https://en.wikipedia.org/wiki/UTF-8#:~:text=Defined%20by%20the%20Unicode%20Standard,Transformation%20Format%20%E2%80%93%208%2Dbit.&text=UTF%2D8%20is%20capable%20of,8%2Dbit)%20code%20units)