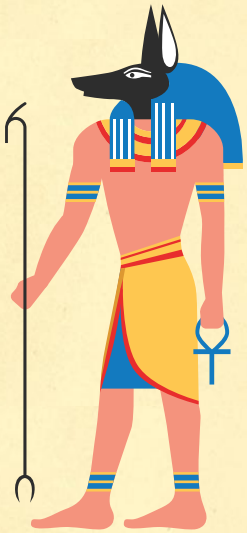
DSA PROJECT

Lahore Garrison University

SENET



An Ancient Egyptian Game

Submitted by

Hamza Asif
342

Mudassir Asif
204

Sadia Khalil
206

Submitted to: Ma’am Sahar Moin

Abstract

This paper is like a treasure map for Senet, an old game from Egypt. We use simple words to talk about what the game is, what you try to do, and how it all works. We also look at why it can be a bit tricky, how to do well, and what Senet can teach us about making other fun games.

Keywords: Senet, Mehen, Hesy-Re, Mastaba, Algorithm

Introduction

- Senet is an ancient Egyptian board game played on a 30-square board. Its earliest representation dates back to around 2620 BCE, found in the Mastaba of Hesy-Re, and similar boards are discovered even earlier in region. It shows 2000 years Egypt's history.
- **Problem:** Game is less efficient & have high time complexity. **Goal:** To make game efficient and reduce time complexity. **I/O:** Players names, Rolling sticks, making move are inputs and winning condition, random score (on rolling stick), checking valid move are outputs if algorithm
- There are many states that control or limits the moves taken by users (further details in constraints section)
- The expected performance of algorithm is to make the gameplay smooth.
- There are multiple test cases that shows time complexity is reduced in various functionalities (excluding total time taken to end the game because it depends on player's moves).

About the Game

1. Objective:

In Senet players have to take their all pawns of the board to successfully win the game.

2. Rules:

While playing Senet following rules are applicable:

[1] Movements:

- Player have to roll the sticks and according to score player will move the pawn.
- If move is valid than user can proceed else player have to make another move. (**Invalid States or moves are discussed in Valid move**)
- Players have to move in ZIG-ZAG format on the board (L →R→L →R)

[2] Interactions:

- Players tries to replace opponent so that they could get ahead of the opponent.

3. Game Elements:

There are total 30 Squares on the Board:

- First 10 squares are reserved for Players pawns.
- 15th Square is Known as House of power.
- 25th Square is known as House of Beauty.
- 26th Square is known as House of Water.
- 27th Square is known as House of Three Spirits.
- 28th Square is known as House of Two Souls.
- 29th Square is known as House of Ones.
- All other squares are set to free.

4. Initial State:

Initially first ten squares of board are occupied by pawns of players (one- after-one) and other 20 squares are set free.

5. Objective State:

When player has successfully taken all the pawns off the board that player wins. To take pawn off the board player needs score that set their position to 31 while keeping validation in consideration.

6. Actions:

Actions that players have to take are:

- Roll the sticks
- Move the pawn
- Try to replace opponent's pawn

7. Move Validation:

Constraints or Invalid Moves:

These are some constraints that makes a move an invalid move:

- Player can't replace opponent's pawn at checkpoint.
- Player can't replace opponent' pawn when there are two or more opponent's pawns in row.
- Player can't replace its own pawn.
- Player can't jump over the house of beauty without getting on house of beauty.
- Player need 3 score to move from house of Three Spirits.
- Player need 2 score to move from house of Two Souls.
- Player need 1 score to move from house of Ones.

- When any player lands in House of water it is moved to House of Power (26 → 15)

8. Transition Function:

The state of game after transition depends on the score and move made by player that changes the position of player's pawns or opponent's pawn (replacement).

9. Game Loop:

When player rolls the sticks and select the pawn to move if move is valid the pawn is moved and then condition is checked if that player wins the game then exit else next player's turn and process goes on.

10. End Conditions:

When player has only one pawn and that player got enough score to take their last pawn off the board wins the game.

11. User Interface:

We have created User Interface using Dart & Flutter (in future we'll work in coding backend for that interface). (Details in slides)

Literature Review:

Senet, an ancient Egyptian board game, dates back to around 3100 BCE, believed to originate during the Predynastic Period. Found in burial tombs, it reflects its role in funerary practices and spiritual beliefs, possibly symbolizing the journey through the afterlife.

There are many games that shares similarities with Senet or have connections to it in terms of culture context or gameplay. Some of these games are:

- Mehen: It is an Ancient Egyptian game that is distinct from Senet.
- Twenty Squares: It is an ancient Egyptian game similar to Senet, but it is not well known as Senet.
- Senat: It is a modern board game that is inspired by Senet, it draws inspiration from Senet.

Problem Statement:

Some problems that might a player has to face are:

- Trying to reach end of board.
- Player can't move when all moves are invalid.
- Player can't replace opponent on checkpoints.

Algorithm Design Objectives

The main objective of the Algorithm design is to make game less time complex and space complex, we have used various methods and functions.

Proposed Solution

We have proposed a solution that helps to enhance the gameplay, by making it smooth and more efficient. Starting with taking names of players as input and then setting board for both players. After, player roll the sticks and choose the pawn to move. Then, it is checked that if move is valid or not, if move is valid then we proceed to check if there is opponent's pawn we have to swap the pawns else simply move player's pawn then we have applied a check-win condition after that 2nd player's turns starts and loop goes on until any player wins the game.

Testing and Iteration:

We have enhanced our algorithm on the bases of test cases and feedbacks of some players. That helps to reduce iterative processes in the algorithm without effecting gameplay.

Proposed System Algorithm (Summarized):

Step 1: Start

Step 2: ***Input*** names of both players

Step 3: ***Set*** board for both players

Step 4: player roll the sticks

Step 5: Player selects the pawn to move

Step 6: ***if*** move is valid

if there is opponent's pawn at new position

replace the opponent pawn

goto **Step 8**

else

move the pawn

goto **Step 8**

else

if player has only one pawn ***OR*** can't move any pawn

print you can't move

goto **Step 8**

else

print choose another pawn to move

goto **Step 5**

Step 7: ***if*** Players all pawns are off the board

print that player wins the game

***goto* Step 9**

Step 8: Next Player's turn

***goto* Step 4**

Step 9: End

SCAN ME 😊 for extended flowchart

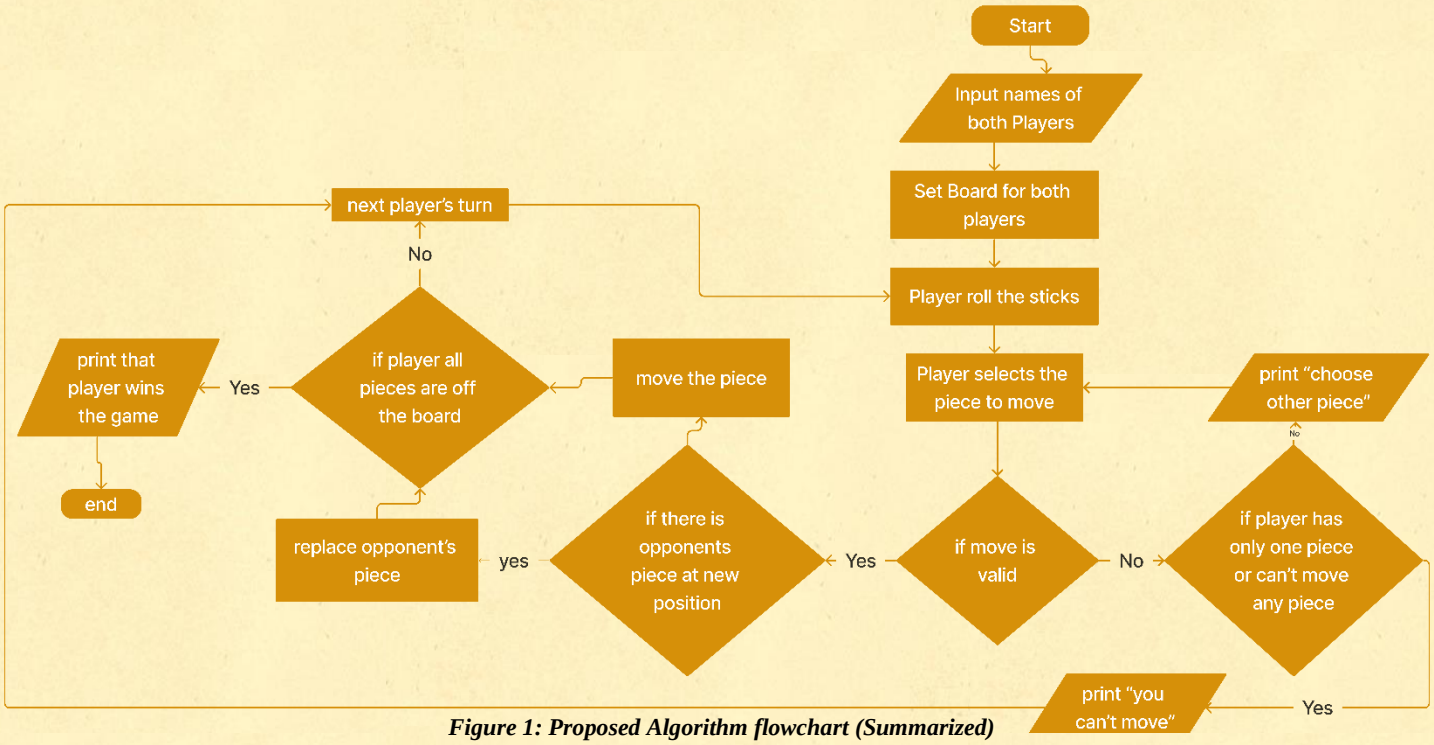


Figure 1: Proposed Algorithm flowchart (Summarized)

Conclusion:

Our algorithm helped to enhance the gameplay, in future we will try to implement it in a real life working project (Game), we have already designed the UI of our game

References

[1] Passing from the Middle to the New Kingdom: A Senet Board in the Rosicrucian Egyptian Museum

[2] Wikipedia: Senet or senat (Ancient Egyptian: , romanized: znt, lit. 'passing'; cf. Coptic ⲥⲓⲛⲉ /sinə/, 'passing, afternoon')