Code / Project : CME1252 / 2
Year / Semester : 2021-2022 Spring Semester
Duration : 5 weeks

Project: Star Trek Warp Wars

The aim of the project is to develop a robot "space maze" game, in which treasures are collected while walking in it.

General Information

The game is played in a 23*55 game field including walls. There are two competitors: Player (P) and Computer (C). There are some treasures/numbers in the game, which the players collect to increase their scores. The aim of the game is gaining the highest end-game score.

Game Elements

P : Player
- Player uses cursor keys to control P.
- Player has a backpack (size of 8 elements).
- Player uses WASD keys to remove an element from the backpack.
- Player has an energy for quick movement (2 times faster).

C : Computer robots
- Computer controls all C robots.
- Treasures are 2 times valuable for computer.
- C robot selects a target, then tries to go to that target directly.

Numbers (**1-5**): Treasure elements
- **1-3** : Static numbers. They cannot move.
- **4-5** : Moving numbers. They move randomly.

Other treasures:
- **=** : Trap device. It stops the numbers and C robots.
- ***** : Warp device. It warps the numbers and C robots (So, they are out of game).
  - Computer robots and numbers cannot detect/avoid trap/warp devices.
  - Trap and warp devices do not affect human player.
  - Trap and warp devices have an effect (trap/warp) area: Square of the device and 8 neighbor squares (total 3*3 area)
  - Trap and warp devices have a duration: They are active for 25 seconds after activation, then they disappear.

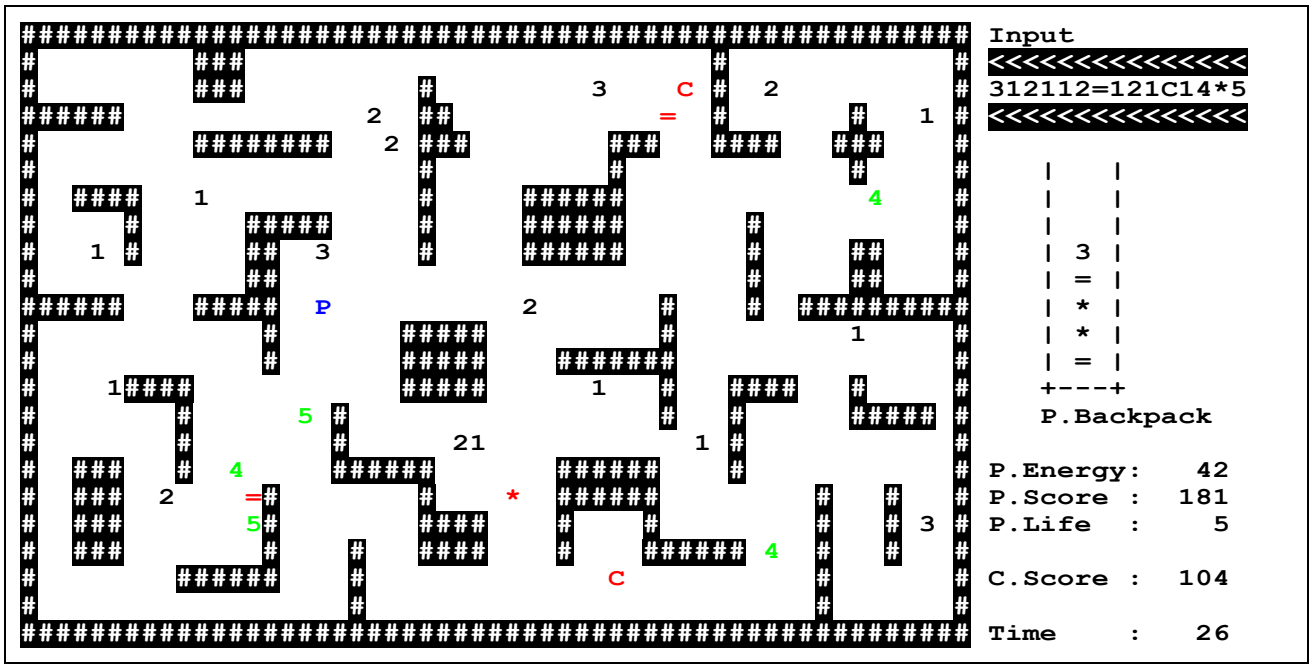
Input Queue

The treasure numbers (1-5), trap (=) and warp (*) devices, C robots are inserted into the maze area from an input queue. The input queue (size of 15 elements) is always full of elements, and shows the next element which will be inserted into the maze. The first element in the queue is inserted into the maze, at a random place in every 3 seconds.

Treasure	Score points for Player	Score points for Computer	Generation probability for Input Queue
1	1	2	12/40
2	5	10	8/40
3	15	30	6/40
4	50	100	5/40
5	150	300	4/40
=	-	300	2/40
*****	-	300	1/40
C	300	-	2/40

Game Initialization

The game area is loaded from a file "maze.txt" at the beginning of the game. The human player (P) and one C robot are placed randomly in the maze. In the beginning, the player has 5 lives and energy for 50 seconds. The first 20 elements of the input queue, are placed into the maze at the beginning of the game for once.



Game Playing Information

The players (P or C) collect the number when they reach to the number's square. Movements are in 4 directions (no diagonal movement). There cannot be more than one game element in the same square. Game elements which cannot take or harm each other, behave like walls when they encounter.

Normal game speed is 2 frames per second for all movements. Player quick movement is 2 times faster than normal speed. When the player has no energy, the player loses the quick movement ability temporarily and can move only at normal speed.

When the human player reaches a number's square, he/she takes the number's score points and also places the number (for 2, 3, 4 or 5) into the backpack. At the top of the backpack, if two elements are identical numbers, they turn into another power or device. If two elements are different numbers, they disappear.

Two identical numbers	turns into Power or Device
2	Energy for 30 seconds
3	Trap device
4	Energy for 240 seconds
5	Warp device

Backpack elements (2, 3, 4, 5, = or *) can be removed from the backpack by using WASD keys. In that situation, removed element is placed into the neighbor square of P (using related direction with the key). If the removed element is a number, it disappears. If removed element is a trap or warp device, it is activated there for 25 seconds. Trap and warp devices are Starfleet technology. So, the human player gains the score of every warped number (static or moving) or warped C robots.

Computer robots have normal speed but treasures are 2 times valuable for score points. If a C robot becomes neighbor with P, P loses 2 elements of backpack. If a C robot reaches the square of P, P loses 1 life. If the human player loses all 5 lives, game ends. When the game ends, end-game score is calculated.

End-Game Score of the Human Player = P.Score - C.Score

Suggested Weekly Program

1. Design of classes, data structures, menu and screen. Load operations.
2. Player movements. Timing. Backpack implementation.
3. Input queue implementation. Static numbers (1-3). Random moving numbers (4-5).
4. Computer movements. Trap device. Warp device.
5. Remaining parts of the game, debugging/testing.

First Evaluation: 11.4.2022 **Report:** 11.4.2022	**Final Evaluation:** 22.4.2022 (powerpoint+poster) **Report:** 22.4.2022