Formal Modeling of a Tetris Game

Mestrado Integrado em Engenharia Informática e Computação

Métodos Formais em Engenharia de Software

Grupo 1 Turma 4MIEIC02
Ângela Cardoso - up200204375
Tiago Galvão - up201500034
Nuno Valente - up200204376

December 24, 2016

Contents

1 Informal system description and list of requirements

1.1 Informal system description

Tetris game it's a puzzle game and one of the most recognizable and influential video game brands in the world. It's no wonder why there are hundreds of millions of Tetris products being played, worn, and enjoyed by fans in their everyday lives. The game was born in 1984 and it's living proof of a game that have truly transcended the barriers of culture and language.

The rules to play the game are very simple. Tetris game requires players to strategically rotate, move, and drop a chaining of tetrominos that fall into the rectangular board at increasing speeds. Players attempt to clear as many lines as possible by completing horizontal rows of blocks without empty space, but if the tetrominos surpass the skyline the game is over! Speed and consequent level advance can make the game ally to strategy more enthusiastic.

One meritorious reference to Alexey Pajitnov because he his the person who developed this popular game. He is a russian video game designer and computer engineer and in his spare time, he drew inspiration from his favorite puzzle board game, pentominos, and decided to create a computer game for himself. Pajitnov envisioned an electronic game that let players arrange puzzle pieces in real time as they fell from the top of the playing field. The resulting design was a game that used seven distinctive geometric playing pieces (appendixB), each made up of four squares. Pajitnov called this game "Tetris," a combination of "tetra" (the Greek word meaning "four") and "tennis" (his favorite sport).

maybe put some image here

1.2 List of requirements

jpojpij

2 Visual UML model

ijiojioj

2.1 Use case model

pojopjop

2.2 Class model

jojijio

3 Formal VDM++ model

dsffds

3.1 Class Game

fdsfds

3.2 Class Board

dfsfds

3.3 Class Cell

dfsfs

3.4 Class Tetramino

sdfds

4 Model validation

fds

4.1 Class MyTestCase

sfds

4.2 Class TestGame

sdfs

5 Model verification

sdfs

5.1 Example of domain verification

sdfs

5.2 Example of invariant verification

sdfs

6 Conclusions

dsfsf

7 References

- https://en.wikipedia.org/wiki/Tetris
- http://tetris.com/

A Source Code

B The 7 tetrominoes