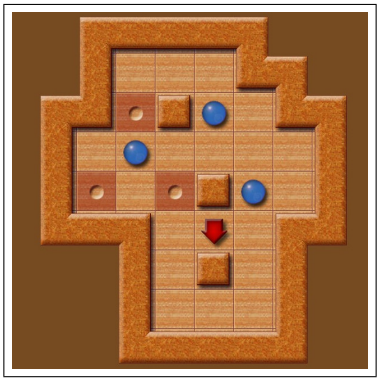
Sokoban puzzle

Research & Development project spring 2019



Deadline: Thursday 25th of April 23:59

Short Problem Description

Sokoban (Warehouse keeper) is a classical puzzle game centered around sliding blocks into correct positions. In this assignment you will be making an app in which you can play puzzles for a variant of Sokoban.

Specification

You have to create an android app with (at least) two screens: a screen to select puzzles, and a screen in which a chosen puzzle can be played. There should be at least 3 (three) playable puzzles.

A puzzle consists of a $n * m$ grid. A puzzle has at least one player character and one victory condition. You should implement the following single tile gameplay elements:

- A player character. Can move 1 tile at a time in any of the cardinal directions.
- Walls. Cannot be moved through.
- Boulders. Can be pushed by the player character.
- At least 1 (one) of the following victory conditions:
  - Boulder slot tiles. To win, the player needs to place a boulder on every slot in the level.
  - A flag/goal tile. When the player reaches this tile, they win.
- At least 1 (one) of the following:
  - Water tiles. Cannot be moved through by the player. When a boulder is pushed into it, it turns into an empty space.
  - Key and door tiles. Doors cannot be moved through by the player. Keys can be pushed like boulders. When a key is pushed into a door, the key disappears and the door turns into empty space.
  - Moving belt tiles. Any object that moves onto a belt is shifted in the direction of the belt.
  - A cool object/tile you come up with yourself.

Figure 1: Examples of the water, key/door and belt elements.

If you still have time left and want to earn a bonus to your grade you can: implement multiple of the optional gameplay elements, and/or come up with your own gameplay elements / improvements. Some example ideas:

- Music / sounds.
- Animations (moving blocks / player, animated tiles)
- Puzzle editor / creation mode.
- Random generation of puzzles.

Handing in your work

As a a final product, you should hand in your documented source code, the .apk file of your app (being the build of your handed in source), and a small document explaining choices you made in design/implemented features (in pdf or plaintext).