Report

In our program we used 2 activities. The first services main menu, and the second game. Menu activities have one text view “Welcome in Sokoban Game” and three buttons : “New Game”, “Rules” and “Exit”.

The second activity starts, when the “New game” button pressed. It will stop worked, when user finished the game and pressed the “No” button.

The most important classes in our program are Level and SokobanView.

In the class Level our program have eight variable and eight functions

Variables:

int numberOfBoxes -stores info about how many boxes is in level

int numberOfGoals -stores info about how many goals is in level

int numberOfRows -stores info about how many rows map have

int numberOfCols -stores info about how many cols map have

int pusherM -stores info about pusher (worker) rows coordinate

int pusherN -stores info about pusher cols coordinate

ArrayList<Character> origGrid -stores the array of original map of level

ArrayList<Character> grid -stores the array with actual map

The symbols which appear in array and their meaning:

“#” -the wall

“$ “ -the box

“*” -the box on goal

“@” -the pusher

“+” -the pusher on goal

“.” -the goal

“ “ -the floor

Functions:

public void restart() is used to restart level. It change the array grid to the array origGrid

public char readPosition(int i, int j) is used to read the character in (i,j) place in the array

public void writePosition(int i, int j, char letter) is used to write character in (i,j) place in the array. It's important to actualize the view of game.

public boolean pusherMove(char direction) is used to check whether the movement of pusher is possible. That function takes variable direction, which stores information about the direction of slide in the screen. Next it check by function readPosition what is in the way of pusher. If there is empty space, marked space or the box with empty/marked space behind, than it returns true, else it returns false.

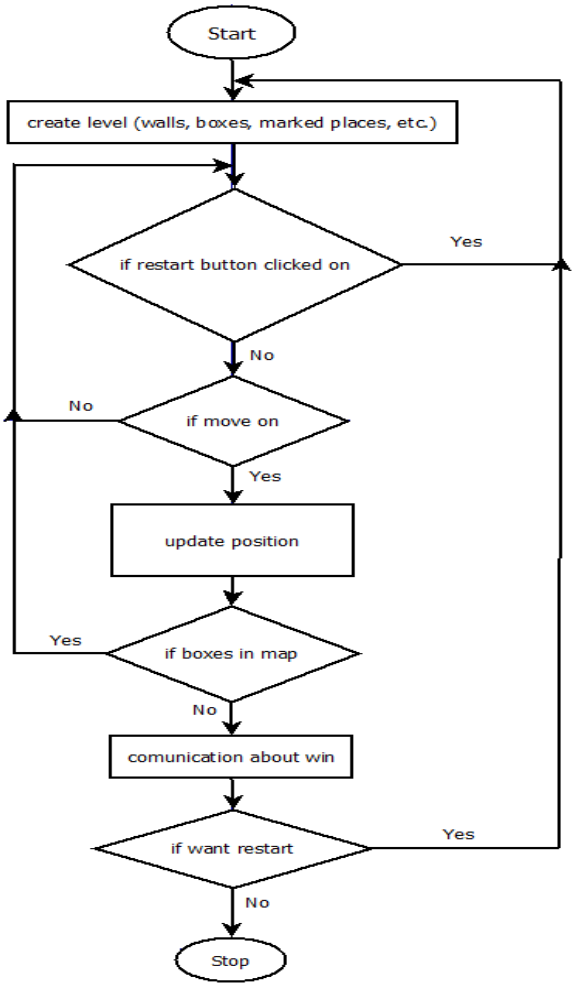
public int move(char direction) is used to change the actual map. It take the same variable than pusherMove(). It used function readPosition and writePosition for change the placement of worker or box.

private void initializePusher() is used to find the pusher in the array and write it's coordinates to pusherN and pusherM variables.

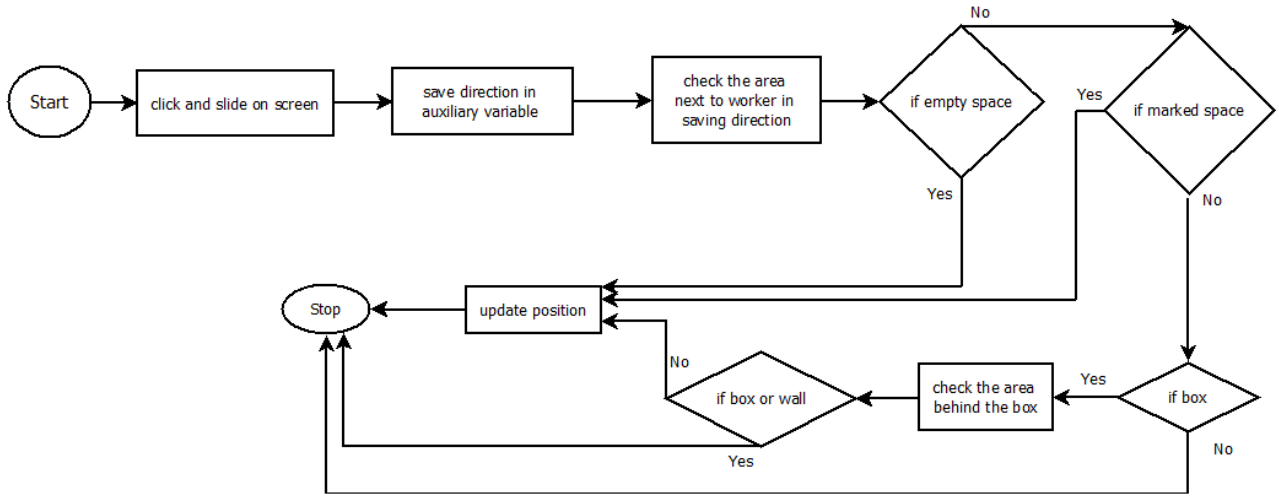
private void makeFloor(), makeFloorPart(int i, int j) is used to make the floor at beginning of game.

public boolean is_won() is used to checked whether victory conditions have been met. If in the array is not $ symbols that means, than all boxes is in goal places. Than it return true. Else it return false.

We present here the game algorithm:



For worker's move, we used the algorithm, that we present in following picture :



In the class SokobanView we have twenty three variables and three functions.

Variables:

int boxSize -stores info about the size of the square with graphics

int buttonSize -stores info about the size of the restart button

int startX -stores info about begin the X coordinate of the box

int startY -stores info about begin the Y coordinate of the box

int restartX -stores info about begin the X coordinate of the restart button

int restartY-stores info about begin the Y coordinate of the restart button

float oldEventX -stores info about beginning X position of finger on the screen

float oldEventY -stores info about beginning Y position of finger on the screen

float oldEventTime -stores info about beginning new event with move the finger on the screen

float distanceX -stores info about actual X position of finger on the screen

float distanceY -stores info about actual Y position of finger on the screen

float speedX, speedY -stores info about speed of the pusher based on the finger move

Bitmap box, boxgoal, goal, wall, restart, pusher, floor, pushergoal -this variables stores graphics for the box, the boxgoal etc.

Level level -stores info about the level (level class which was described earlier)

RectF rect -stores info about rectangles by which the map is build

Functions:

void onDraw(Canvas canvas) is used to return actually view. It used class level to check the elements in the array and display adequate graphics.

void onSizeChanged(int width, int height, int oldWidth, int oldHeight) is used to giving to class variables appropriate values.

boolean onTouchEvent(MotionEvent event) is used to input operating. It checking (by class level) is winning condition true. If yes than set the message about win and ask the question about restart or end. Else it checked is the move by slide on screen, or click on the restart button.