Synopsis.

We chose the shop window of The Game Box in Enschede. Our choice was based on the function of this store/entertainment facility. The Game Box is an arcade hall where you can play lots of games. Our main idea was to animate and program some games in the store front. In our animation, the logo, the round logos and the main window come back, which will be described in the usage and interactions.

Usage and interactions

We made 3 games in our shop front window.
Game 1: Game one tests your reaction time. When you click too early you fail, when you click on the right moment you get a notification on your response time.

Game 2: The second game is a game in which you need to move the character, with the arrow keys, to the stars as fast as possible. When you hit 20 stars the game ends and you get prompted with a score that's your time in milliseconds divided by 100.

Game 3: In our third game, we created a 2d version of a rubix cube, as this is what the logo of the Game Box represents. When clicking on one of the 9 cubes, the color will change randomly. You can play around with different patterns of color combinations or just make it all the same color. In total, there are six different colors. When all the colors are the same, the window will switch from game 1 to game 2

Architecture

Our most important functions that are used are the ones to actually play the game. The standard functions as setup and draw are also worth mentioning but we believe that these are common for every program. In our program, we use mouseClicked and keyPressed to play these games and to let our program function.

Let's start with mouseClicked. This is used for two of the three games. Firstly, the reaction time game, seen in the windowshop. As described before, this game tests your reaction time. Secondly, the 2d rubix cube. Per cube, you can click to change the color. When you have completed the game, which means, getting all the colors the same, you will move on to game 3. Game 3 is shown in the same window as game 1 and is, as described above, a game to collect stars as fast as possible. The mouseClicked function is needed to play game 2.
For game 3, keyPressed is used, as this game is played with the arrows on your keyboard.

We created three classes, one for each game. Each class creates the display that is needed for the game, and codes the action that is needed to play the game. In our main tab, the references to the classes are mentioned in the functions keyPressed and mouseClicked.

Structure (diagram)