"La Sapienza" University of Rome
Faculty of information engineering, information technology and statistics
Department of informatics, automation and control engineering
"ANTONIO RUBERTI"
Degree program: Artificial Intelligence and Robotics

HOMEWORK №2

Student: OREL, Egor
Matricola: 1836231
Cloud storage containing the solution: https://yadi.sk/d/DUZNDkKTfWn1Kw

Teacher: SCHAERF, Marco

Rome, 2019

Exercise: to obtain the following effects:

1. Create a hierarchical model of a (simplified) horse, composed of the following parts:
   a. a body;
   b. 4 legs, each one composed of 2 independent components (upper and lower leg);
   c. a head;
   d. a tail.

   All components are cubes, use the cube function present in the file

2. Add a procedural texture to the body of the horse. The texture should be a checkerboard pattern but with a linear decrease of intensity from the front to the back of the body. Use, as a reference, textureCube4 of Chapter 7 of the examples of the textbook. Notice however that you should not apply a sinusoid but a linear decrease in the direction of the tail.

3. Create a (very simplified) model of a jump obstacle made with cubes.

4. Add a button that starts an animation of the horse so that, starting from an initial position where it is standing and positioned along the x axis, it walks by moving (alternatively back and forth) the legs, then jumps above the obstacle and lands after it.

Results

1) A draft model of an animal was given. I decided to add a tail and a neck to make the object equal to a real horse. Final hierarchy of the horse body is presented at Figure 1. For this purpose, changes were made inside of InitNodes(Id) function (lines 290 - 302), function for drawing a neck and a tail were added (lines 475 - 488) and a number of NumNodes was increased from 11 to 13 (line 177).
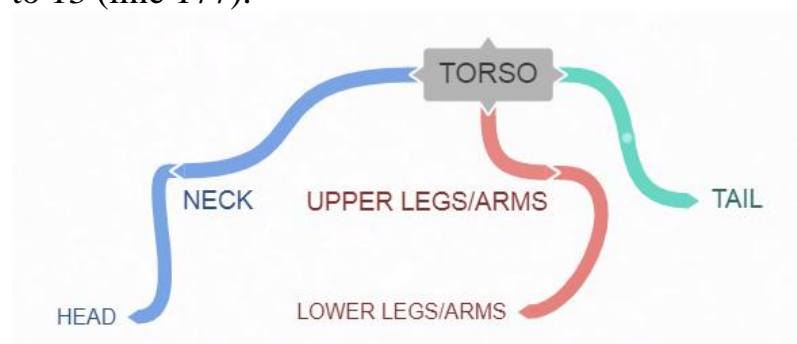

Figure 1 – Horse body hierarchy

2) The texture of the torso and of other parts of the body are different (Figure 2). The linear pattern is applied on legs, neck, head, and tail (lines 51 – 62 of the *.js file). On the torso, I applied a checkerboard with a linear decrease of the color in the color in direction front→back (this decrease is defined in the block of code lines 36 – 49 of *.js file). To make the horse more beautiful, I applied also a color gradient onto the body (lines 35 – 43 of the *.html file).

Figure 2 – Body of the horse

3) The jump obstacle is made as a second hierarchical tree, independent from the horse body (lines 218 – 270, 644 of the *.js file). The structure of the obstacle is presented on the Figure 3. Texture of the obstacle is equal to the body of the horse (except checkerboard structure) + the track with checkerboart texture is added.
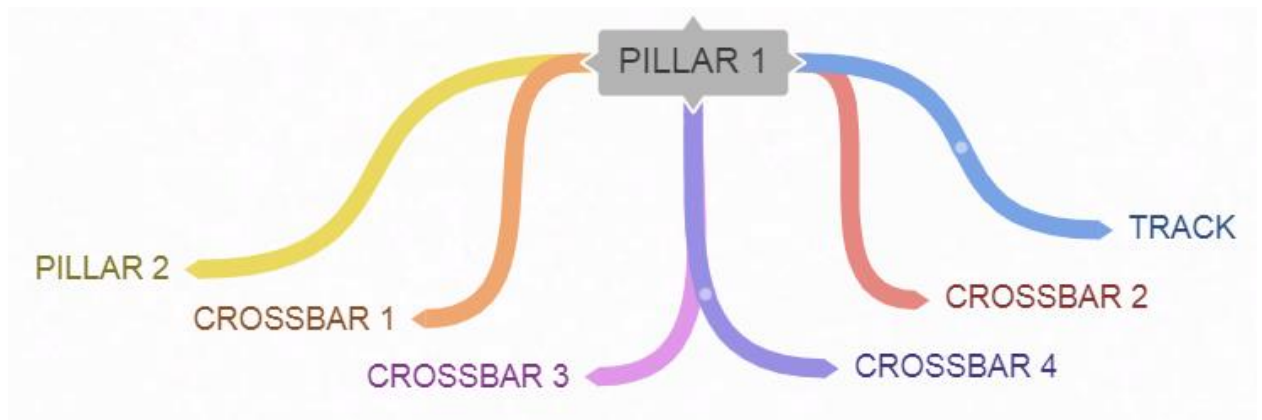

Figure 3 – Hierarchy of the obstacle

4) Animation of the horse is managed using different functions for:
   a. upper parts of legs and arms → runUpperLegsPosition();
   b. lower legs position → runLowerLegsPosition();
   c. lower arms position → runLowerArmsPosition();
   d. torso horizontal and vertical movement → runTorsoPosition().

All the animation process is created in lines 654 – 656, 673 – 757 and specified in the render() function of the *.js file.

Also there are two buttons to control the animation on the screen: the first reloads the page to the initial settings, the second is a PLAY/PAUSE toggle for animation.

Movements of all body parts are specified using sinusoidal dependency to make it smoother. All the way of the horse is divided to 3 main parts: running before the obstacle(1), jumping above the obstacle(2), running and stopping after the obstacle(3). At the (2) distance segment amplitude of sinusoud is

3

increased and speed of legs rotation is decreased to imitate body grouping of the horse during the jump. Then at (3) it smoothly continues.

<u>Advantages of the proposed solution:</u>
- camera position can be changed by the user changing positions of Theta and Phi sliders;
- horse and obstacle are independent hierarchical structures, so it is possible to "play" with them separately;
- thanks to a hierarchical structure of bodies it is possible to control animation of all children changing only parameters of parents.

<u>Disadvantages of the proposed solution:</u>
- movements and body structure of the horse could be more naturalistic;
- animation could be cycled to jump without pauses in forward and backward directions, for example. Now a user has to reload the program to start the animation from the initial position;
- horse and obstacle could be coloured by some different colors and textures to make it more beautiful.