Snake

# Змия

Snake е стара класическа видео игра. Създадена е края на 70 години. По-късно е адаптирана и за компютри. Играча управлява змията. Целта е да се изядат възможно най-голям брой ябълки. Всеки път когато змията изяде ябълка, тялото и се удължава. Змията трябва да избягва стените и собственото си тяло.

# Разработка

Размера на всяка една от частите на змията е 10px. Змията се управлява със стрелките. Първоначално дължината на змията е 3 части. В края на играта се изписва “Game Over” .

Първата ни задача е да дефинираме константите, които използваме в нашата игра.

**private** **final** **int** boardWidth = 800;

**private** **final** **int** boardHeight = 600;

**private** **final** **int** snakeWidth = 10;

**private** **final** **int** maxDotsNumber = boardWidth \* boardHeight / snakeWidth / snakeWidth; //max number parts of snake

**private** **final** **int** timeDelay = 100;

С константите boardWidth и boardWidth задаваме размерите на екрана за игра. snakeWidth е размера на ябълката и всяка една от частите на змията. maxDotsNumber дефинира максималния брой възможни части, които змията може да достигне. timeDelay определя скоростта на играта

**private** **int** x[] = **new** **int**[maxDotsNumber]; //the snake position for each part

**private** **int** y[] = **new** **int**[maxDotsNumber]; // in x[0] & y[0] set position for head

x[] и y[] –тук се пазят координатите на змията на частите от тялото.

x[0] и y[0] са координатите на главата.

**private** **int** snakeLength=3; //start length of snake

**private** **int** appleX; //position for apple

**private** **int** appleY;

В snakeLength задаваме първоначалната дължина на змията.

appleX и appleY се пазят координатите на ябълката.

**private** Image img = **new** ImageIcon("backgroundImg.jpg").getImage();//background image

**private** Image snakePart = **new** ImageIcon("dot.png").getImage(); //image for snake part

**private** Image apple = **new** ImageIcon("apple.png").getImage(); //image for apple

**private** Image snakeHead = **new** ImageIcon("headRight.png").getImage();//image for snake head

С тези променливи задаваме картинките в играта – фона, частите от змията, ябълката и главата на змията. Използваме класа ImageIcon за изобразяване на картинките.

**public** SnakeGameMachine() {

addKeyListener(**new** CheckKeyPressed());

setBackground(Color.***BLACK***);

setFocusable(**true**);

setPreferredSize(**new** Dimension(boardWidth, boardHeight));

**for** (**int** i = 0; i < snakeLength; i++) {

x[i] = 400 - i \* 10;

y[i] = 200;

}

newApple();

timer = **new** Timer(timeDelay, **this**);

timer.start();

}

В SnakeGameMachine метода създаваме змията, ябълката на случайно място и стартираме таймера.

**private** **void** checkEatApple() {

**if** ((x[0] == appleX) && (y[0] == appleY)) {

Sound.*play*("sounds/bloop.wav"); //plays bloop.wav when eating apples

snakeLength++;

newApple();

}

}

**private** **void** newApple() {

**int** r = (**int**) (Math.*random*() \* (boardWidth / 10 - 1));

appleX = ((r \* snakeWidth));

r = (**int**) (Math.*random*() \* (boardHeight / 10 - 1));

appleY = ((r \* snakeWidth));

}

В метода checkEatApple() проверяваме дали има сблъсък между ябълката и главата на змията, в случай на колизия се чува звук, увеличаваме дължината на тялото, и извикваме метода newApple(), който позиционира нова ябълка на случайно място.

**public** **class** Sound {

**public** **static** **synchronized** **void** play(**final** String fileName) {

**new** Thread(**new** Runnable() {

**public** **void** run() {

**try** {

Clip clip = AudioSystem.*getClip*();

AudioInputStream input = AudioSystem.*getAudioInputStream*(**new** File(fileName));

clip.open(input);

clip.start();

} **catch** (Exception e) {

System.***out***.println("play sound error");

}

}

}).start();

}

}

За да използваме звуци в играта имаме класа Sound. Работи с файлове с разширение .wav.

За да е възможно свиренето на повече от един звук по едно и също време, създаваме нова нишка за всеки звук (намират се в папка "sounds" в папката на проекта). При възникнала грешка при изпълнение на звука, се изписва "play sound error" в конзолата.

**private** **void** move() {

**for** (**int** i = snakeLength; i > 0; i--) {

x[i] = x[(i - 1)]; //this code moves the parts

y[i] = y[(i - 1)];

}

**switch** (direction) {

**case** 'L':

x[0] -= snakeWidth;

**break**;

**case** 'R':

x[0] += snakeWidth;

**break**;

**case** 'U':

y[0] -= snakeWidth;

**break**;

**case** 'D':

y[0] += snakeWidth;

**break**;

**default**:

**break**;

}

}

В метода move() е алгоритъма за клавишите на играта. Контролираме главата на змията, можем да сменяме посоките със стрелките. Останалата част от змията се премества с по една позиция след главата. Втората част се премества там където е била първата, третата където е била втората и т.н.

**private** **void** checkGameOver() {

**for** (**int** i = snakeLength; i > 0; i--) {

**if** ((i > 4) && (x[0] == x[i]) && (y[0] == y[i])) {

GameOver = **true**;

}

}

**if** ((y[0] >= boardHeight) || (y[0] < 0) || (x[0] >= boardWidth) || (x[0] < 0)) {

GameOver = **true**;

}

**if** (GameOver) {

Sound.*play*("sounds/game\_over.wav"); //plays game\_over.wav when GameOver

timer.stop();

}

}

В метода checkGameOver(), проверяваме дали змията се е ударила в някоя от стените или в себе си. При край на играта се пуска звук Game Over и се изписва съобщението на екрана.