CREATING THE CLASSIC GAME TETRIS

Goals:
My goal of this project was to complete the code for a fully functioning Tetris game. I wanted to first create the basic concepts of the game: make the tetrads move down, have them stay at the bottom of the screen, generate a new tetrad, have the game end when necessary, and then have them clear once a full line of blocks is created. From there I wanted to be able to take the project further, by creating better graphics, a scoreboard, a line counter, and having the speed increase with more lines cleared. I also wanted to incorporate bonus points after a certain number of lines, as well as a screen on the right in which phrases would pop up after you cleared a certain number of lines.

Method:
The first thing I had to do was look over the codes given to me to see what I needed to do. It was pretty evident that the Tetris.java class needed the most work. The Tetris class already had methods for creating new tetrads, using the keyboard for the game, drawing the border and background, and clearing the board after each movement of the tetrad and what not.

The first thing I had to do was enable the falling movement for the tetrads. I was able to do this by incrementing the initialized “cn” variable, which in turn called the tetrad to move down, at certain increments. This ended up being a success, but the tetrads would not stay at the bottom, and would simply disappear. After looking over the Tetromino code, I found a method “stay”, which I was able to use after the code tested the tetrad for the “atBottom” condition. The “createNewTetrad” function was then implemented, so that the process would repeat again.

From here I approached my next problem, the ability to clear lines. I did this by creating first a “checkLines” function (1), and then a “Removecompleletelines” function (4). The first checked for a full line, called on the second function that would clear the line to 0 and move the above lines down. From here I implemented the “gam0ver” function (5) into the program, which was a Boolean that would be activated in the main function. When all this was done, I had the basics of the game Tetris. The next step was to implement extra components to enhance the gaming experience.

My next step was creating a scorekeeper (3) as well as a line counter (2) that were relatively simple functions implemented into the remove line function. I used an “if” statement within the score class to generate bonus points as well as a speed regulator in the line counter class to make the game more difficult as more lines are cleared. From here I had felt as though I had implemented enough components to make the game entertaining. So now it was just a matter of graphics, which were imposed by using “draw.java” as well as copying a tetrad function to shadow the
blocks. Making it appear as though it is outlined. I also was able to have certain phrases pop up on the side when a certain number of lines were cleared. All in all I would call the game a success.

**Short Comings:**
The biggest problem was the speed regulator. Which for one reason or another at first went up in moderation, but once 12 lines were cleared would increase tetrad movement at a ridiculous rate. I tried to change this by how the “speed variable” (y) was manipulated, but I couldn’t get it to make that smooth transition, although it still did increase gradually in speed when the first line was cleared.

The other shortcoming was the music. Which for some reason or another, I couldn’t get “StdAudio.java” to work properly, even after tracing the .wav files path.

One component I didn’t finish was the line multiplier, which gave more points for clearing numerous lines at once. I started this function, but didn’t have the time to make it function accurately.

I had a lot of fun with this project, and didn’t mind putting in the hours as the game began to come together. In the future I could expand on the project more by adding the multiplier and music.

```java
1) public void checkLines()
    {
        for(int i=0; i<8H; i++){
            for(int j=0; j<8W; j++){
                if(board[i][j]==0)
                    break;
                if(j==8W-1 & board[i][j]!=0){
                    Removecompleteline(i);
                    draw.show();
                }
            }
        }
    }

2) public void linescleared()
    {
        if(y>3)
            if(z==1 & y<200){
                y = y-4;
            }
    }

3) public void addscore(){
    x = x+10;
    if(x>=100 & x<200){
        x = x+10;
    }
}

4) public void Removecompleteline(int line){
    for(int i = line; i > 0; i--){
        for(int j=0; j<8W; j++){
            board[i][j]=board[i-1][j];
        }
        addscore();
        linescleared();
    }
}

5) public boolean gameOver(){
    int i = 0;
    for(int j=0; j<8W; j++){
        if(board[i][j]==0 & board[i][j]<9 & i<25){
            createNewTetrad();
            draw.show(5);
            draw.clear();
            return true;
        }  
    }
    return false;
}
```
Main Function:

```java
public static void main(String[] args) {
    Tetris tetris = new Tetris();
    tetris.createNewTetrad();
    int cn = 0;
    while(true) {
        tetris.drawBorder();
        tetris.clearBoard();

        tetris.tetrad.move();
        if(cn%y == 0)
            tetris.tetrad.moveDown();

        if(!tetris.tetrad.canMoveDown()) {
            tetris.tetrad.stay();
            tetris.checkLines();
            tetris.createNewTetrad();
            if (tetris.gameOver()) {
                tetris.gameoversong();
                tetris.createNewTetrad();
                tetris.draw.show(5);
                tetris.Done();

                break;
            }
        }
    }
}
```

Screenshots:
Edom's Tetris Game

Score: 10
Line: 1
Noob!

Tetris
GAME OVER

FINAL SCORE: 0