

Gebze Technical University
Department of Computer Engineering
CSE 241/501
Object Oriented Programming / Programming
Fall 2022
Homework # 3
Dynamic Memory and Classes
Due date Nov 30th 2022

You will convert your Tetris game of HW2 into a version that uses **dynamic memory** with **new** and **delete** operators. If you don't have one working we will provide one. You will **not use** any STL classes in this homework such as vectors.

Your class will again be named **Tetris**. It will have at least the following **functions**

- **Constructors** to take the rectangular size of the Tetris board.
- The **operator +=** to add a Tetromino to the board. The new Tetromino will be **added at the top row in the middle.**
- **No fit function** will be implemented!!
- **Draw function** to draw the Tetris board. It will optionally start the drawing from the top. See how to move your cursor to the top of your screen on a UNIX terminal at https://en.wikipedia.org/wiki/ANSI_escape_code#CSI_sequences. Note that your code will not be portable because it will work only on certain consoles.
- **Animate function** to animate the added tetromino dropping to the bottom of the board. The animation will be repetition of four steps:
 1. Draw the board with Tetromino at the top
 2. Ask the user **rotation direction** and rotation count
 3. Ask the user **move direction** and count
 4. Rotate and move the Tetromino
 5. **Draw** the board
 6. Sleep 50 milliseconds
 7. Lower the Tetromino one level and **go to step 5** until it hits the bottom.

You will submit two driver source files, **each will include a main function**. Your **first** driver code will **test** each member function of both **classes (Tetris and Tetromino)** and **print** (or show) the result on the screen.

Your **second** driver code will do the following

1. Ask the user the **size** of the Tetris board
2. Ask the Tetromino **type** (I, O, T, J, L, S, Z). User may enter **R for random** Tetromino, **Q** for quit.
3. Add the asked Tetromino to the board and animate
4. Go to **2**

Notes:

- Define your **namespace** for the classes

- You will submit 2 header and 2 CPP files for two classes. You should also submit 2 driver CPP files named driver1.cpp and driver2.cpp
- You will submit the makefile of your program. Submission without makefile will not be evaluated.
- Do not use any functions from the standard C library (like printf), do not use C arrays. For math functions you may use standard C functions.
- You will not use any STL classes in this homework such as vectors.
- Your program should have header file and implementation files
- Use all the OOP techniques that we have learned in the lectures such as consts, C++11 features (range for loops, strong enums, auto keyword, decltype keyword, etc.)
- Do not forget to indent your code and provide meaningful comments.
- Check the validity of the user input.
- **Test your programs very carefully at least with 5 different runs. For some runs use trivial cases such as 3 O tetriminos.**
- You should submit your work to the Teams page using the instructions from the TAs.