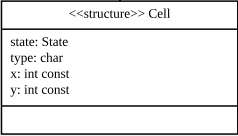
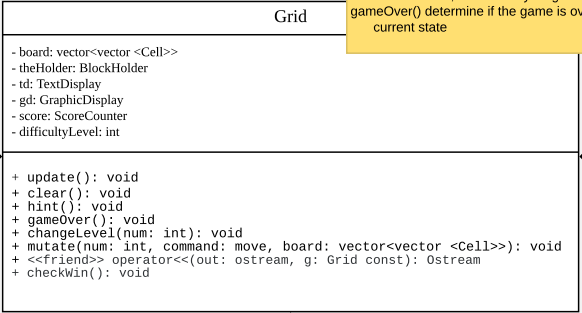
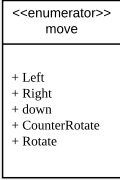
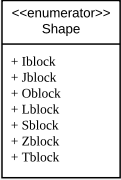
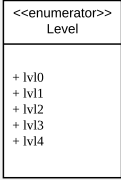
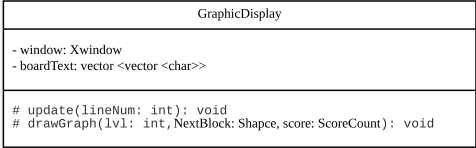
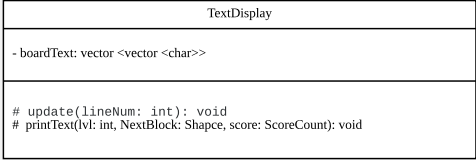
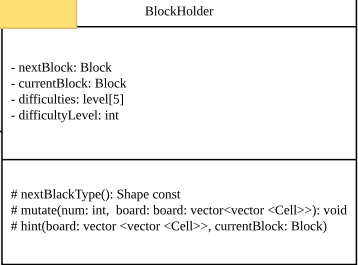


the class count the score



Grid
The largest class that contents others as field;
functions:
hint() call the hint() in "BlockHolder", print the best solution
clear() clear the board
changeLevel() change the difficult lvl of the game
update() check if need to delete any fuled lines
--- if it does, (1) it call getScore() in score
(2) call update() in "BH"
--- if it doesnt, it will do anything
gameOver() determine if the game is overafter that, it will print the current state



BlockHolder
- this class manage the next block and current block
- the cells covered by currentBlock should be State::temp
functions:
NewBlock(): prepare the next block for the game,
this functon should be run when user let function "drap"
mutate(): mutate the block and the board
update(): mutate num to 0

