Gebze Technical University
Department of Computer Engineering
CSE 241/505
Object Oriented Programming
Fall 2019
Homework # 5
Inheritance
Due date Dec 12th 2019

In this homework, you will design and implement a class hierarchy for the Board class of the NPuzzle problem. You will also write global functions to use these classes.

Your **`AbstractBoard`** class represents the board configuration as you did in the previous homework assignments.

Fuction Name	Explanation
`print`	Prints the board on the screen by sending it to cout
`readFromFile`	Reads the board from the file given as function parameter. The file format is defined as in HW2.
`writeToFile`	Writes the board to the file given as function parameter
`reset`	Resets the board to the solution.
`setSize`	Sets the board size to given values. The values are given as parameters and there are no restrictions on the board size. The board is reset after this operation.
`move`	Makes a move according to the given char parameter. If the parameter is ‘L’ then the blank tiles moves left, …, etc, as defined in HW1.
`isSolved`	Returns true if the board is a solution
`Operator()`	Takes two indexes and returns the corresponding cell content. Terminates program if the indexes are not valid.
`Operator==`	Two boards are equal, if the boards are the same. This operator does not consider last move or the number of steps
`NumberOfBoards`	Returns the number of Board objects created so far.
`lastMove`	Returns the last move, if there is no last move, returns ‘S’
`numberOfMoves`	Returns the number of steps (moves) this board made

Many of the functions above cannot be implemented because your do not know how the board is represented in this abstract base class. You will derive 3 new concrete classes from this class that represents the boards in different ways:

- **`BoardVector`**: The Board is represented using an STL vector of STL vectors.
- **`BoardArray1D`**: The Board is represented using a one dimensional dynamic C array.
- **`BoardArray2D`**: The Board is represented using a two dimensional dynamic C array.

Write global function that takes an array of **`AbstractBoard`** pointers and returns true if the array contains a valid sequence of moves for a solution (same question of the midterm exam).

Notes:

- Do not use any C++ features that we did not learn during the lectures. Use all the OOPL rules we learned in the class.
- Make your own namespace, use separation of interface and implementation rules.
- Test each function of each class at least once by writing driver code.
- Test the global function at least 5 times with different number of types of board pointers.
- You should submit your work to the moodle page and follow all the submission rules that will be posted.