Gebze Technical University
Department of Computer Engineering
CSE 241/505
Object Oriented Programming
Fall 2020
Homework # 5
Inheritance and Templates
Due date Jan 18th 2021

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 **`AbstractHex`** 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 beginning.
`setSize`	Sets the board size to given values. The values are given as parameters. The board is reset after this operation. Size should be larger than 5x5 and it should be square.
`play`	The class will have two functions named **play** that plays the game for a single step. First function does not take a parameter and it plays the computer. The second function takes a cell position and it plays the user. If the parameter is as defined in HW3.
`isEnd`	Returns true if the board is a game end
`Operator()`	Takes two indexes and returns the corresponding cell content. Throws an object of an exception class of your design.
`Operator==`	Two boards are equal, if the boards are the same. This operator does not consider last move or the number of steps
`lastMove`	Returns the last move, if there is no last move, throws exception.
`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 2 new concrete classes and 1 new templated class from this class that represent the boards in different ways:

- **`HexVector`**: The Board is represented using an STL vector of STL vectors.
- **`HexArray1D`**: The Board is represented using a one dimensional dynamic C array.
- **`HexAdapter`**: This class takes a template parameter and behaves like an adaptor class just like the stack or queue class of the STL. The parameter classes can be any STL class with a random access iterator.

Write global function that takes an array of **`AbstractHex`** pointers and returns true if the array contains a valid sequence of moves for a game.

Notes:

- Use all the OOPL rules we learned in the class.
- Throw exceptions and write code to test them.
- 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.