Gebze Technical University
Department of Computer Engineering
CSE 241/501
Object Oriented Programming / Programming
Fall 2023
Homework # 1
Your First C++ Program
Due date Dec 3 2023

You will write a C++ program that will play the game of (not so smart) chess. Your chess board will be printed on the console and will look like

here in this representation:

K = King
Q = Queen
R = Rook
N = Knight
B = Bishop
P = Pawn

The lowercase letters represent the black pieces, and the uppercase letters represent the white pieces. The dots represent empty squares. Each square is identified by its coordinates, with the files labeled from "a" to "h" and the ranks labeled from "1" to "8".

Here are your programs features

- The program should alternate turns between white and black.
- Implement error handling to address common issues such as invalid input, illegal moves, and other relevant errors.
- Display appropriate error messages and allow the user to correct their input.
- Accept user input for moves in the standard chess notation (e.g., "e2e4").
- Validate the input to ensure it conforms to the standard notation.
- Provide suggestions for legal moves based on the current board position.
- Consider the rules of chess and the specific legal moves for each piece.

Example Interaction

```
Welcome to the Chess Game!

8 | r  n  b  q  k  b  n  r
7 | p  p  p  p  p  p  p  p
6 | .  .  .  .  .  .  .  .
5 | .  .  .  .  .  .  .  .
4 | .  .  .  .  .  .  .  .
3 | .  .  .  .  .  .  .  .
2 | P  P  P  P  P  P  P  P
1 | R  N  B  Q  K  B  N  R
    ----------------------
    a  b  c  d  e  f  g  h

[White's Turn]
Enter your move: e2e4

[Updated Board]
8 | r  n  b  q  k  b  n  r
7 | p  p  p  p  p  p  p  p
6 | .  .  .  .  .  .  .  .
5 | .  .  .  .  .  .  .  .
4 | .  .  .  .  P  .  .  .
3 | .  .  .  .  .  .  .  .
2 | P  P  P  P  .  P  P  P
1 | R  N  B  Q  K  B  N  R
    ----------------------
    a  b  c  d  e  f  g  h

[Black's Turn]
Enter your move: suggest
Suggestion is e7e5

[White's Turn]
Enter your move:…
```

Your C++ program should have the following features

- A class for chess board
- A class for pieces
- The pieces are kept in a std::vector
- A score function that calculates the overall goodness score
    - Pawn: 1 point
    - Knight: 3 points
    - Bishop: 3 points
    - Rook: 5 points
    - Queen: 9 points
    - Safety of each piece: minus half the point of that piece if under attack
- The next move function should return the board that produces best score
- Functions for saving and loading from files.

Notes:

- 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.
- Use C++ standard classes such as string, vector, list, etc.

- 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.
- **Test your programs very carefully at least with 5 different runs.**
- You should submit your work to the Teams page using the instructions from the TAs.
- You will demo your homework online.
- You will submit all your source code with Makefile. The homeworks without Makefile will not be evaluated.