CS 341:Programming Language Design And Implementation

Jon A. Solworth

Fall 2023

Project	1 version 2
Due	**11:59 PM**, Sep. 23rd
Policy	Individual work only, no late submissions
Assignment	Chessboard container and interator
Submission	gradescope

1 Containers and iterators in C++

Write a C++ program with a 8x8 (i.e. 2-dimensional matrix) chessboard and an iterator over that chessboard. *The container and iterator must be constructed from scratch, that is, not using existing containers or iterators.*

The pieces on the chess board have a Color, Piece, and position. The Color is either Black or White. The Piece is one of the following:

- Rook
- Knight
- Bishop
- Queen
- King
- Pawn

We suggest you use enums for both Piece and Color.

The position is given by an x,y coordinate, each coordinate on the board is in the range of 0...7. When the chessboard is set up to play a game, White's pieces are in rows 0 and 1, while Black's pieces are in rows 6 and 7.

1.1 The container

The container Chessboard should have the methods

- Chessboard() the constructor creates an empty chessboard, that is one with no pieces on it.
- int place(int x, int y, Color c, Piece p): to place the piece on the chessboard. It should return
  - 1 if successful,
  - -1 if illegal x coordinate
  - -2 if illegal y coordinate
  - -3 if square is already occupied
  - -4 if illegal color
  - -5 if illegal piece
- int get(int x, int y, Color &c, Piece &p): to get the value of the piece on the chessboard.
  - 1 if successful,
  - -1 if illegal x coordinate
  - -2 if illegal y coordinate
  - -3 if square is not occupied
- int move(int fromX, int fromY, int toX, int toY): to move piece fromX, fromY to toX, toY
  - 1 if successful,
  - -1 if illegal fromX coordinate
  - -2 if illegal fromY coordinate
  - -3 if illegal toX coordinate
  - -4 if illegal toY coordinate
  - -5 if fromX,fromY is not occupied
  - -6 if toX, toY is occupied with a piece of the same color as the one being moved
  - -7 if illegal move *other than -6*

  We consider only simple moves, and thus do not include castling, en-passant, or a pawn becoming another piece through reaching the last row.

  Note that move includes capturing a piece which is then removed from the chessboard.
- print() Print the 8x8 board using the iterator.

  In a board which is in the initial configuration for chess, the white pieces should be on the bottom and the black pieces the top. Use 3 characters to print each square:

" . " *(space period space) if unoccupied*

cp *(color piece space) if occupied where*

- *Color is b or w for Black or White.*
- *Piece is R, N, B, K, Q, P for Rook, Knight, Bishop, King, Queen, and Pawn.*

Here is an example of the initial starting position for a game of chess:

```
bR bN bB bQ bK bB bN bR
bP bP bP bP bP bP bP bP
 .  .  .  .  .  .  .  .
 .  .  .  .  .  .  .  .
 .  .  .  .  .  .  .  .
 .  .  .  .  .  .  .  .
wP wP wP wP wP wP wP wP
wR wN wB wQ wK wB wN wR
```

1.2 Iterator

The iterator, nameed ChessboardIterator, should go through the chessboard in the order

$$[0,0],[1,0],\ldots,[7,0],[0,1],[1,1],\ldots$$

Where $[x, y]$ is the x,y position on the board.

There should be a method which translates the iterator into x,y coordinates for the chessboard.

- int it.xy(int &x, int &y)
  - *returns 1 if it refers to a place on the board*
  - *returns -1 if it is not on the board*

1.3 Files

Your should submit 3 files.

- A main.cpp file with tests of your program
- A chessboard.cpp file with all your code for the classes.
- A chessboard.h file which is included in main.cpp and chessboard.cpp

We will test your program with our own main.cpp and we will release a main.cpp example for you to use.

Electronic Submission

Before submission, make sure your name appears somewhere on your assignment. When you are ready to submit, login to Blackboard, find Assessments in the left hand side, then follow the link to Gradescope (or navigate to Gradescope directly), and then submit your image to "Project 1". You may submit as many times as you want, but we grade only the last submission.

Policy

All work is to be done individually — group work is not allowed. While we encourage you to talk to your peers and learn from them, this interaction must be superficial with regards to all work submitted for grading. This means you *cannot* work in teams, you cannot work side-by-side, you cannot submit someone else's work (partial or complete) as your own. The University's policy is available here:

`https://dos.uic.edu/conductforstudents.shtml` .

In particular, note that you are guilty of academic dishonesty if you extend or receive any kind of unauthorized assistance. Absolutely no transfer of program code between students is permitted (paper or electronic), and you may not solicit code from family, friends, or online forums. Other examples of academic dishonesty include emailing your program to another student, copying-pasting code from the internet, working in a group on a homework assignment, and allowing a tutor, TA, or another individual to write an answer for you. It is also considered academic dishonesty if you click someone else's iClicker with the intent of answering for that student, whether for a quiz, exam, or class participation. Academic dishonesty is unacceptable, and penalties range from failure to expulsion from the university; cases are handled via the official student conduct process described at `https://dos.uic.edu/conductforstudents.shtml` .