

Central Washington University
College of the Sciences
Department of Computer Science
CS-301 Data Structures Fall 2016

Project 3

As usual, please read the “Guidelines” document on how to turn in this project. Please note:

- Due date is on a Tuesday rather than on the usual Wednesday.*
- See the special provisions, which include slight modification to Guidelines.*

*Due date: **Tue Nov 29, 2016 11:59 pm***

1. A mating program. This program will may involve several data structures and the use of recursion. Your program is going to read a file `position.fen` that contains a few chess positions using with W (White) to move using the Forsyth-Edwards notation (FEN) (check the details in Wikipedia or elsewhere. Also explore PGN notation for chess notation). Your output must be on a file `solutions.txt` which must contain:
 - A diagram of the position
 - *All* the sequences of moves(at most 5) leading to B (Black) being mated or stating `No 3-move mates found`.

A sample input and output will be provided soon.

The objective of the program, is to find at most 3-move check mates in certain type of easy problems. Ideally, your program should implement an `Application`

```
class CheckMateIn3 implements Application
```

and use the class `BackTrack` to solve the problem. However, other approaches allowed, including some tweaking. In games in general, backtracking must do exhaustive search (within constraints): finding *a mate* is not guarantee of success, as it depended on a possible B wrong move.

The problems expected as input are *easy* in the sense that the search trees are pre-pruned:

- Every choice (called *ply* in this context) is a check for W,
- Consequently, if B is to move, it is a check-parrying ply.
- The tree cannot be too deep: W mates on the 5th ply worst case.
- No unusual moves like castling or en passant are allowed.

Note: a well implemented program should easily handle more than 3 move limit, but the size of the data structures may become too large.

Special Provisions

Please consider this before turning your project:

- These are going to be three person projects, with a leader as usual, but I will emphasize that "Each member is responsible for the whole project".
- In the cover sheet, include a section call *Additional Notes* specifying details of the implementation, data structures of for the state, whether you are using the *Application* interface or implemented ad-hoc backtracking, etc. Not more than one page but mention references if needed.