4) There is a 5 point bonus if you hand it a discussion report of at least 1 full page discussing and answering the following questions:

1. You will notice that often the time limit set at GUI is surpassed in runtime. Why do you think it is?
2. Any suggestion how to remedy that so we can have a time limit that is respected by the runtime?

1. The time is only tracked in the minmax_player and alpha_beta_player functions is why the time almost always go over the GUI time set. There is a while loop running in both above functions, that while loop checks if the time or the largest depth possible has been passed. Since that is the only place the time is checked the runtime can co over the time limit by how ever long it take to finish the current loop. Since each loop checks a lower and lower depth each successive loop take exponentially longer, meaning you can run overtime quiet a lot.

```
def alpha_beta_player(game, state):
    start = time.perf_counter()
    end = start + game.timer
    """use the above timer to implement iterative deepening using alpha_beta_cutoff() version"""
    move = None

    while time.perf_counter() < end and game.d <= len(state.moves):

        #print("Your code goes here -10pt alpha_beta_player")

        print("iterative deepening to depth: ", game.d)
        move = alpha_beta_cutoff(game, state, game.d)
        game.d = game.d + 1

    return move
```

```
def minmax_player(game, state):
    start = time.perf_counter()
    end = start + game.timer
    """use the above timer to implement iterative deepening using minmax_cutoff() version"""
    move = None

    while time.perf_counter() < end and game.d <= len(state.moves):

        # print("Your code goes here -10pt minmax_player")

        print("iterative deepening to depth: ", game.d)
        move = minmax_cutoff(game, state, game.d)
        game.d = game.d + 1

    return move
```

2. To solve this problem, you would need to check the current time more often. You could pass the “end” variable to the functions being called and if time is ever greater the “end” stop the loop there and pick the best result you have so far.