Name: _____ **CCID:** _____

1. **[4 points]** In the example code `mcts/mcts1.py`:

   (a) What is the difference between `get_best_move` and `best_uct`?

   (b) Consider the following code from the `best_uct` method:

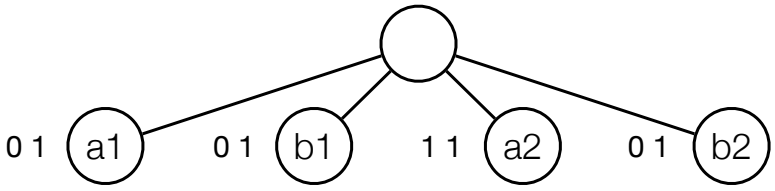
   ```
   # calculate UCT, update if best
   mean_res = child.results / child.sims
   uct = ___(1)___ + (self.c * sqrt(log(self.root_node.sims) / child.sims))
   if best_uct is None or uct > best_uct:
       best_uct, best_child = uct, child
   ```

   What should missing expression (1) be?

   (c) `self.c` gets initialized to 0.3. What would be the effect of setting it to something much larger (e.g., 50)?

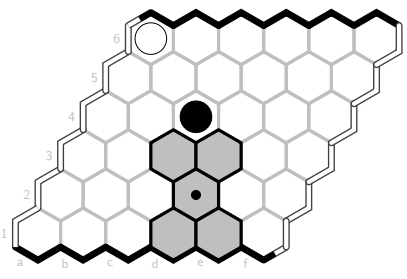
   (d) If you run `mcts/main.py` multiple times and ask for a winning move for Black (using the command `g x`), it will sometimes generate different moves in different runs. Why?

2. **[3 points]** Below is the Monte Carlo search tree after the first 4 simulations of running the `g x` command of `mcts/main.py` with a 2×2 board. Show the tree after simulation 5 (output below); explain each change. The indices map to positions as 5=`a1`, 6=`b1`, 9=`a2`, 10=`b2`.




```
trv_xpnd bu * .4 .4 1.4 .4 9
  xpnd_nd * 9 > 5
  xpnd_nd * 9 > 6
  xpnd_nd * 9 > 10
sim 5. * 9 5 roll 6 parent loss
```

3. **[5 points]** Consider the following virtual connection in 6×6 Hex:



   (a) Does this depict a *full connection* or a *semi-connection* between the Black stone and the bottom border? Explain your reasoning.

   (b) Write this connection in logical form as an AND/OR strategy.

   (c) To follow this strategy, where must Black play after `1.B[c4] 2.W[a6] 3.B[d2] 4.W[e1]`?