0. Cutoffs
Provide a short diagram or description to explain the following parameters from P3:

(a) **ply**

**Solution:**
The total number of levels ahead looked (so, the height of your search tree).

(b) **cutoff**

**Solution:**
The number of levels remaining in the tree when your search switches to a non-parallel algorithm.

(c) **divideCutoff**

**Solution:**
The maximum number of threads which should be forked sequentially when dividing-and-conquering a list of moves. Similar to the `sequentialCutoff` parameter from exercises.

(d) **PERCENTAGE_SEQUENTIAL**

**Solution:**
The maximum percentage of a list of moves which should be forked sequentially in order to determine reasonable values for alpha and beta.

1. Efficiency
Circle the most efficient option from each pair of possible implementation strategies for P3:

(a) To create threads for each move in a `List<M>` during Parallel Minimax:

- Create threads in a for loop  **OR**  Create threads with divide-and-conquer

**Solution:**
Create threads with divide-and-conquer.

(b) To pass copies of boards to these threads:

- Copy the board inside the thread  **OR**  Copy the board before passing it to the thread

**Solution:**
Copy the board inside the thread.

(c) To evaluate a list of moves using Alpha-Beta pruning:

- Evaluate the moves in the order provided  **OR**  Sort the moves best-first, then evaluate in sorted order
Solution:
Sort the moves. Sorting is fast, and allows us to prune more effectively by establishing tight Alpha/Beta bounds.

2. Alpha-Beta
Determine the value of the root node after running Alpha-Beta on the following tree (and cross out pruned branches/nodes):

```
Max's Turn

X

Min's Turn

A 30
B 40

30 50 20

Max's Turn

Y

Min's Turn

C 2
D 4
E 2
F 4

3 2 3 4

Z

Max's Turn

Min's Turn

Solution:

Max's Turn

Min's Turn

Max's Turn

Min's Turn

Max's Turn

Min's Turn

β = 40

β = 4