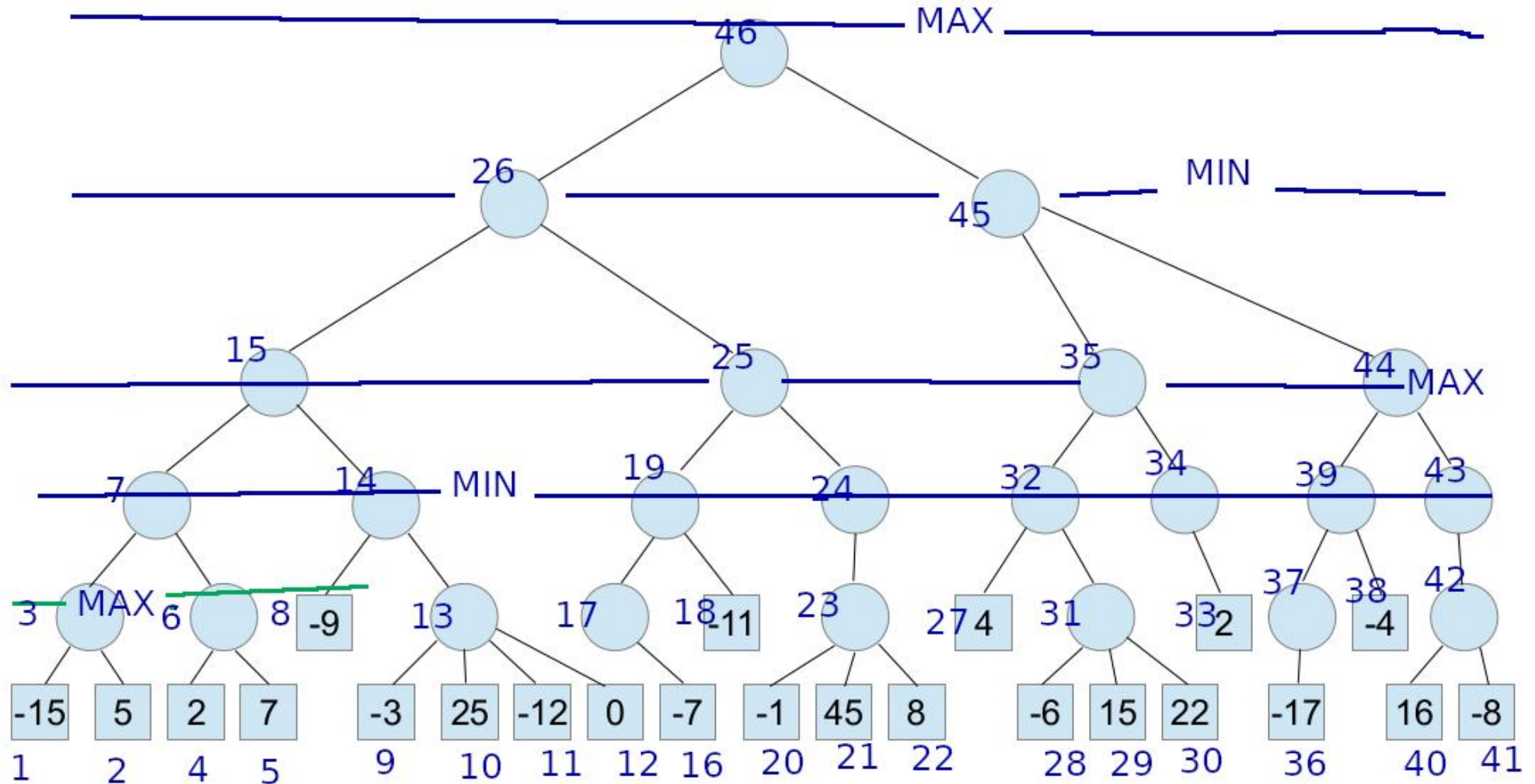


You will be working on the MiniMax tree shown above.

Assume: 2 Players, player 1 moves first. Square nodes are terminal nodes.

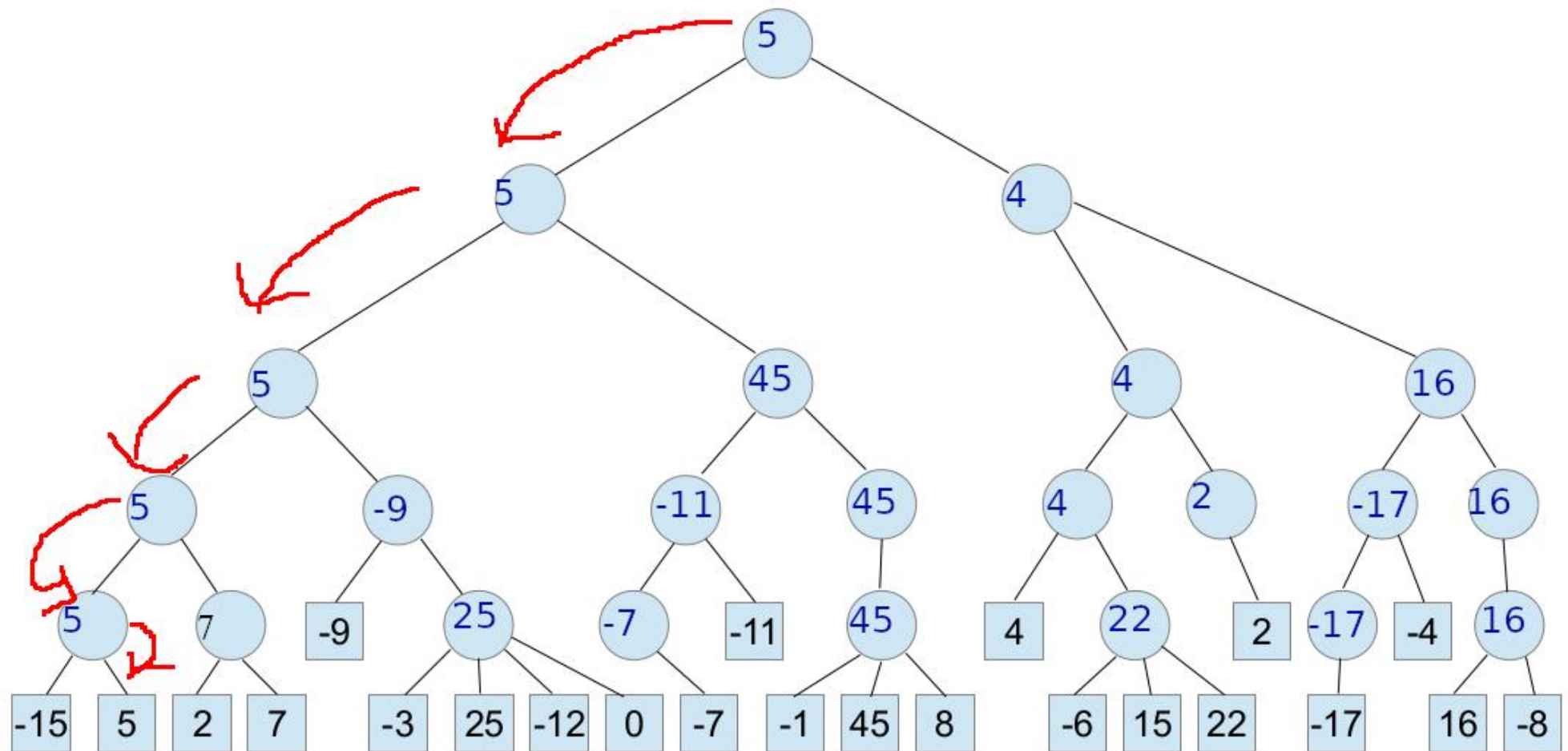
Remember: MiniMax is a DFS search, ***you must work in correct DFS order or things won't turn out as you would like.*** For nodes at the same level, order of expansion is ***left to right.***



Q1 - On the tree above, write next to each node (including terminals) the **order** in which the node is **expanded** by the MiniMax search.

Q2 - For each level of the tree, indicate whether nodes are **MAX** nodes or **MIN** nodes. How many **complete turns** are there?

TWO (2)



Q3 - Write **within each node** the correct **MiniMax** score for the node. Do this in the same order in which that score would be computed by the search procedure.

Q4 - Mark the path corresponding to the actions that would be selected by each player all the way to a terminal node. Who wins the game?

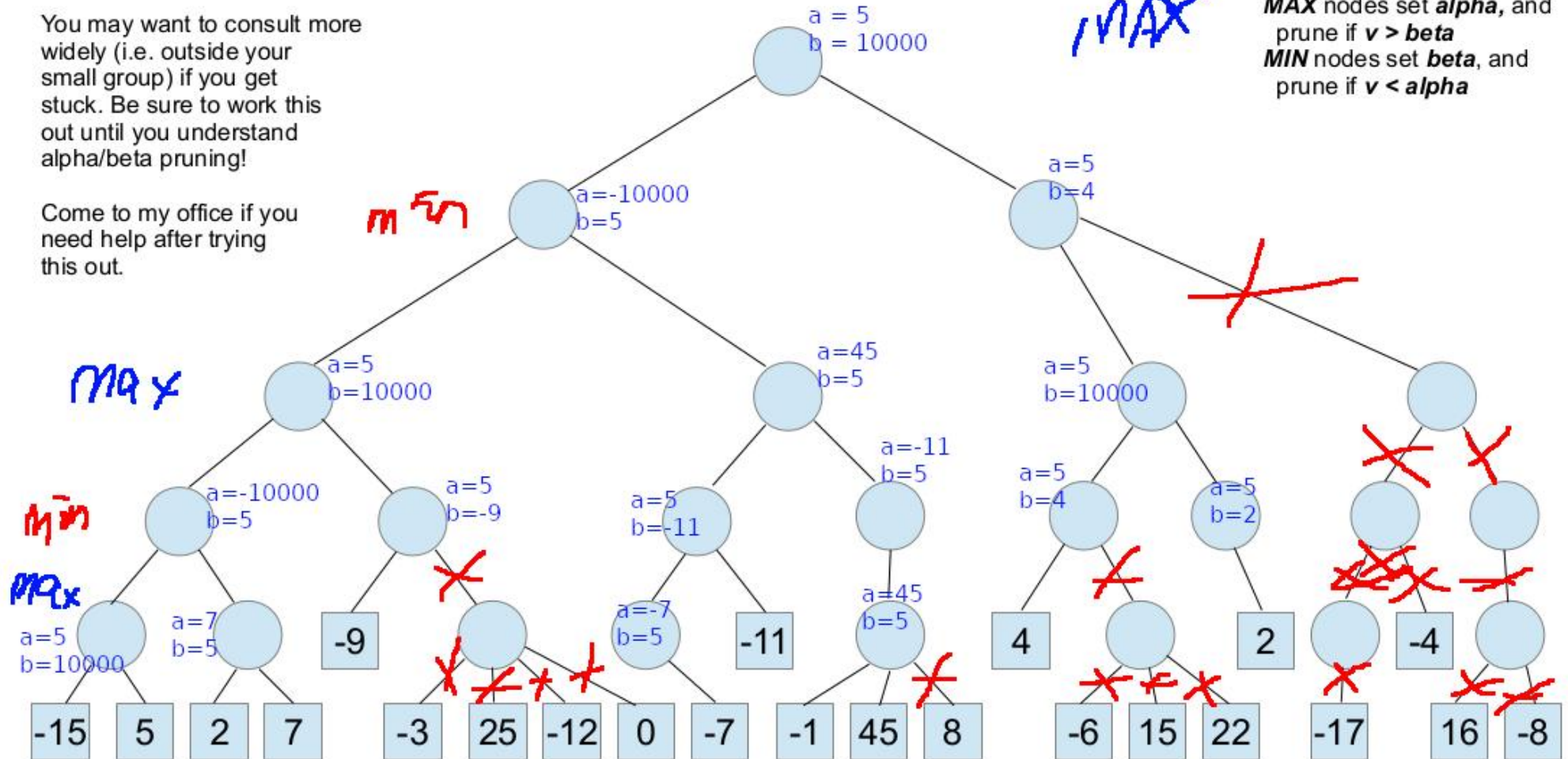
Player 1 win the game



You may want to consult more widely (i.e. outside your small group) if you get stuck. Be sure to work this out until you understand alpha/beta pruning!

Come to my office if you need help after trying this out.

Remember:  
**MAX** nodes set **alpha**, and  
 prune if  $v > \text{beta}$   
**MIN** nodes set **beta**, and  
 prune if  $v < \text{alpha}$



Q5 – This is the tricky but very important part:

Using the correct **DFS search order**, the **MiniMax scores** for each node, and assuming **alpha=-10000, beta=10000** initially for the root node:

Mark which nodes/branches are pruned by ***alpha/beta pruning***. And note the alpha or beta value that caused the pruning to take place.

Name: Zhang Jinming

Student number: 1000997503

Practice by hand using the website, and implement a simple version in python and check the result, then finish the assignment to implement a real version of alpha beta pruning.