1. (25 pts total, -5 pts for each error, but not negative) MINI-MAX SEARCH IN GAME TREES.

The game tree below illustrates a position reached in the game. Process the tree left-to-right. It is Max’s turn to move. At each leaf node is the estimated score returned by the heuristic static evaluator.

1.a. Fill in each blank square with the proper mini-max search value.

1.b. What is the best move for Max? (write A, B, or C) __________

2. (25 pts max, -5 for each error, but not negative) ALPHA-BETA PRUNING. Process the tree left-to-right. This is the same tree as above (1.a). You do not need to indicate the branch node values again.

Cross out each leaf node that will be pruned by Alpha-Beta Pruning.
3. **(50 pts total, 10 pts each)** Execute Tree Search through this graph (i.e., do not remember visited nodes, so repeated nodes are possible). It is not a tree, but pretend you don’t know that.

Step costs are given next to each arc. Heuristic values are given next to each node (as \( h = x \)). The successors of each node are indicated by the arrows out of that node. Successors are returned in left-to-right order. **(Note: C is a successor of itself).**

For each search strategy below, indicate the order in which nodes are expanded (i.e., to expand a node means that its children are generated), ending with the goal node that is found. The first one is done for you as an example.

3.a. **DEPTh FIRST SEARCH.**

\[
\text{S A D G1}
\]

3.b. **(10 pts, -3 for each wrong answer, but not negative)** UNIFORM COST SEARCH.

3.c. **(10 pts, -3 for each wrong answer, but not negative)** GREEDY (BEST-FIRST) SEARCH.

3.d. **(10 pts, -3 for each wrong answer, but not negative)** ITERATED DEEPENING SEARCH.

3.e. **(10 pts, -3 for each wrong answer, but not negative)** A* SEARCH.

3.f. **(10 pts, -3 for each wrong answer, but not negative)** OPTIMALITY.

Did Uniform Cost Search find the optimal goal? \[ \] Why or why not? 

Did A* Search find the optimal goal? \[ \] Why or why not?