Exercise 3.1 (Board Games)

(a) Consider the game tree for the two-person board game depicted below. Simulate the behavior of the Minimax algorithm with α-β pruning (always expand children from left to right). Enter the computed node values into the triangles and all intermediate α-β values into the appropriate tables.
(b) Consider the problem of search in a three-player game (you may assume that no alliances are allowed) without the zero-sum condition. The players are called 1, 2, and 3. Unlike in the case of two-player zero-sum games, the evaluation function now returns a triple \((x_1, x_2, x_3)\) such that \(x_i\) is the value the node has for player \(i\).

Complete the game tree given below by annotating all interior nodes and the root node with the backed-up value triples.

\[
\begin{align*}
\text{Sp. 1} & \quad \text{Sp. 2} & \quad \text{Sp. 3} \\
(1, 2, 3) & \quad (4, 2, 1) & \quad (6, 1, 2) \quad (7, 4, -1) & \quad (5, -1, -1) & \quad (-1, 5, 2) \quad (7, 7, -1) \quad (5, 4, 5)
\end{align*}
\]

(c) Assume that the value triple \((5, 4, 5)\) at the rightmost leaf node is replaced by \((5, 4, -1)\). Which problem arises now when you try to back up value triples? Suggest how to modify the back-up procedure to obtain a “robust” result at the root node.

**Exercise 3.2** (Satisfiability, Models)

(a) Decide for each of the following propositions whether they are valid, satisfiable or neither valid nor satisfiable.

1. \(\text{Smoke} \Rightarrow \text{Smoke}\)
2. \(\text{Smoke} \Rightarrow \text{Fire}\)
3. \((\text{Smoke} \Rightarrow \text{Fire}) \Rightarrow (\neg \text{Fire} \Rightarrow \neg \text{Smoke})\)
4. \((\text{Smoke} \Rightarrow \text{Fire}) \Rightarrow ((\text{Smoke} \land \text{Heat}) \Rightarrow \text{Fire})\)
5. \(\text{Spring} \Leftrightarrow \text{SunnyWeather}\)

(b) Consider a vocabulary with only four propositions, \(A, B, C,\) and \(D\). How many models are there for the following formulae? Explain.

1. \((A \land B) \lor (B \land C)\)
2. \(A \lor B\)
3. \((A \leftrightarrow B) \land (B \leftrightarrow C)\)
Exercise 3.3 (CNF Transformation, Resolution Method)
The following transformation rules hold, whereby propositional formulae can be transformed into equivalent formulae. Here, $\phi$, $\psi$, and $\chi$ are arbitrary propositional formulae:

\[
\neg\neg\phi \equiv \phi \quad (1) \\
\neg(\phi \lor \psi) \equiv \neg\phi \land \neg\psi \quad (2) \\
\phi \lor (\psi \land \chi) \equiv (\phi \lor \psi) \land (\phi \lor \chi) \quad (3) \\
\neg(\phi \and \psi) \equiv \neg\phi \lor \neg\psi \quad (4) \\
\phi \land (\psi \lor \chi) \equiv (\phi \land \psi) \lor (\phi \land \chi) \quad (5)
\]

Additionally, the operators $\lor$ and $\land$ are associative and commutative.

Consider the formula $((C \land \neg B) \leftrightarrow A) \land \neg C \rightarrow A$.

(a) Transform the formula into a clause set $K$ using the CNF transformation rules. Write down the steps.

(b) Afterwards, using the resolution method, show whether $K \models (\neg B \rightarrow (A \land C))$ holds.

Exercise 3.4 (DPLL)
Use the Davis-Putnam-Logemann-Loveland (DPLL) procedure to find a satisfying assignment for the formula $\phi$. Write down all steps carried out by the algorithm during the process. If you have to apply a splitting rule, split on variables in alphabetical order, trying true first, then false. Indicate the satisfying assignment.

$\phi = (\neg A \lor C \lor \neg D) \land (A \lor B \lor C \lor \neg D) \land (\neg A \lor \neg E) \land \neg C \land (A \lor D) \land (A \lor C \lor E) \land (D \lor E)$

The exercise sheets may and should be worked on in groups of three (3) students. Please write all your names and the number of your exercise group on your solution.