Q1. CSPs with Preferences

Let us formulate a CSP with variables A, B, C, D, and domains of $\{1, 2, 3\}$ for each of these variables. A **valid assignment** in this CSP is defined as a complete assignment of values to variables which satisfies the following constraints:

1. B will not ride in car 2.
2. A and B refuse to ride in the same car.
3. The sum of the car numbers for B and C is less than 4.
4. A's car number must be greater than C's car number.
5. B and D refuse to ride in the same car.
6. C's car number must be lesser than D's car number.

(a) Draw the corresponding constraint graph for this CSP.

Although there are several valid assignments which exist for this problem, A, B, C and D have additional "soft" preferences on which value they prefer to be assigned. To encode these preferences, we define utility functions $U_{Var}(Val)$ which represent how preferable an assignment of the value(Val) to the variable(Var) is.

For a complete assignment $P = \{A : V_A, B : V_B, ...D : V_D\}$, the utility of P is defined as the sum of the utility values: $U_A(V_A) + U_B(V_B) + U_C(V_C) + U_D(V_D)$. A higher utility for P indicates a higher preference for that complete assignment. This scheme can be extended to an arbitrary CSP, with several variables and values.

We can now define a modified CSP problem, whose goal is to find the valid assignment which has the maximum utility amongst all valid assignments.

(b) Suppose the utilities for the assignment of values to variables is given by the table below

U	U_A	U_B	U_C	U_D
1	7	10	200	2000
2	6	20	300	1000
3	5	30	100	3000

Under these preferences, given a choice between the following complete assignments which are valid solutions to the CSP, which would be the preferred solution.

- ○ A:3 B:1 C:1 D:2
- ○ A:3 B:1 C:2 D:3
- ○ A:3 B:1 C:1 D:3
- ○ A:2 B:1 C:1 D:2

To decouple from the previous questions, for the rest of the question, the preference utilities are not necessarily the table shown above but can be arbitrary positive values.

This problem can be formulated as a modified search problem, where we use the modified tree search shown below to find the valid assignment with the highest utility, instead of just finding an arbitrary valid assignment.

The search formulation is:

- State space: The space of partial assignments of values to variables
- Start state: The empty assignment
- Goal Test: State X is a valid assignment
- Successor function: The successors of a node X are states which have partial assignments which are the assignment in X extended by one more assignment of a value to an unassigned variable, as long as this assignment does not violate any constraints
- Edge weights: Utilities of the assignment made through that edge

In the algorithm below $f(node)$ is an **estimator of distance** from *node* to *goal*, ACCUMULATED-UTILITY-FROM-START(*node*) is the sum of utilities of assignments made from the *start-node* to the current *node*.

```
function ModifiedTreeSearch(problem, start-node)
    fringe ← Insert(key : start-node, value : f(start-node))
    do
        if IsEmpty(fringe) then
            return failure
        end if
        node, cost ← remove entry with maximum value from fringe
        if Goal-Test(node) then
            return node
        end if
        for child in Successors(node) do
            fringe ← Insert(key : child, value : f(child) + Accumulated-Utility-From-Start(child))
        end for
    while True
end function
```

(c) Under this search formulation, for a node X with assigned variables $\{v_1, v_2....v_n\}$ and unassigned variables $\{u_1, u_2, u_3...u_m\}$

(i) Which of these expressions for $f(X)$ in the algorithm above, is guaranteed to give an optimal assignment according to the preference utilities. (Select **all** that apply)

☐ $f_1 = \min_{Val_1, Val_2, ... Val_m} U_{u_1}(Val_1) + U_{u_2}(Val_2) + ... + U_{u_m}(Val_m)$

☐ $f_2 = \max_{Val_1, Val_2, ... Val_m} U_{u_1}(Val_1) + U_{u_2}(Val_2) + ... + U_{u_m}(Val_m)$

☐ $f_3 = \min_{Val_1, Val_2, ... Val_m} U_{u_1}(Val_1) + U_{u_2}(Val_2) + ... + U_{u_m}(Val_m)$ such that the complete assignment satisfies constraints.

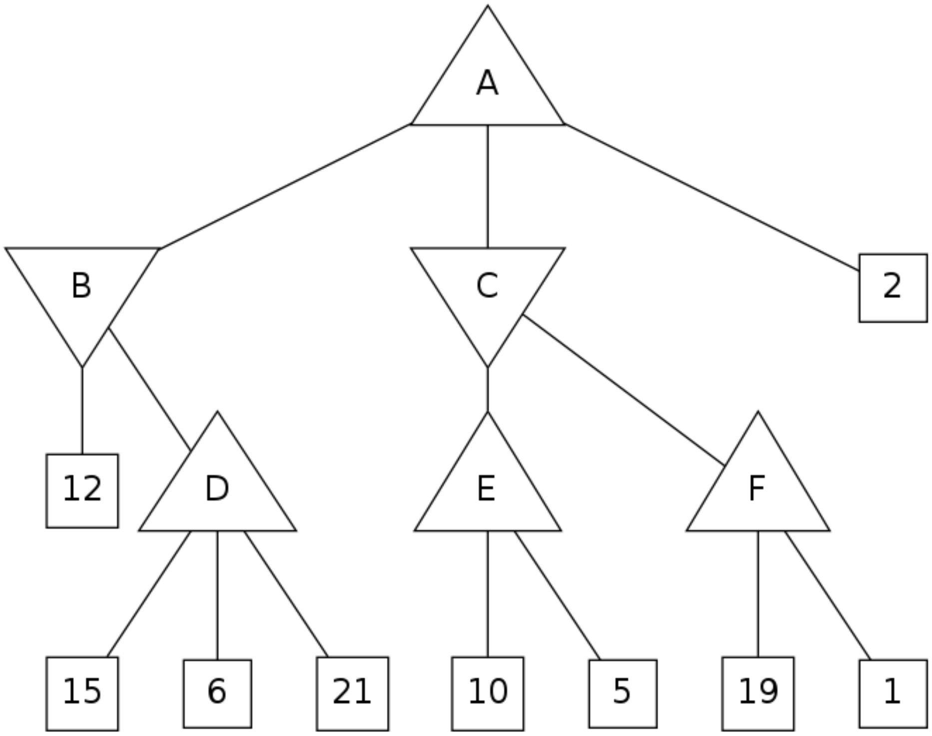
☐ $f_4 = \max_{Val_1, Val_2, ... Val_m} U_{u_1}(Val_1) + U_{u_2}(Val_2) + ... + U_{u_m}(Val_m)$ such that the complete assignment satisfies constraints.

☐ $f_5 = Q$, a fixed extremely high value ($\gg$ sum of all utilities) which is the same across all states

☐ $f_6 = 0$

(ii) For the expressions for $f(X)$ which guaranteed to give an optimal solution in part(i) among $f_1, f_2, f_3, f_4, f_5, f_6$, order them in ascending order of number of nodes expanded by ModifiedTreeSearch.

Q2. Games



(a) What is the minimax value of node A in the tree above?

(b) Cross off the nodes that are pruned by alpha-beta pruning. Assume the standard left-to-right traversal of the tree. If a non-terminal state (A, B, C, D, E, or F) is pruned, cross off the entire subtree.

(c) If a function F is strictly increasing, then $F(a) < F(b)$ for all $a < b$ for $a, b \in \mathbb{R}$. Consider applying a strictly increasing function F to the leaves of a game tree and comparing the old tree and the new tree.

Are the claims below true or false? For true cases, justify your reasoning in a single sentence. For false cases, provide a counterexample (specifically, a game tree, including terminal values).

In a *Minimax* two player zero-sum game, applying F will not change the optimal *action*.

True False

In a *Minimax* two player zero-sum game, applying F will not affect which nodes are pruned by alpha-beta pruning.

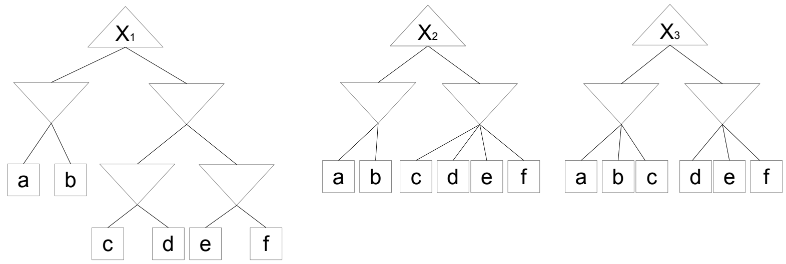
True False

In a *Minimax* two player non-zero-sum game (where the utilities of players do not necessarily add up to zero), applying F will not change the optimal *action*.

True False

In an *Expectimax* two player zero-sum game, applying F will not change the optimal *action*.

True False

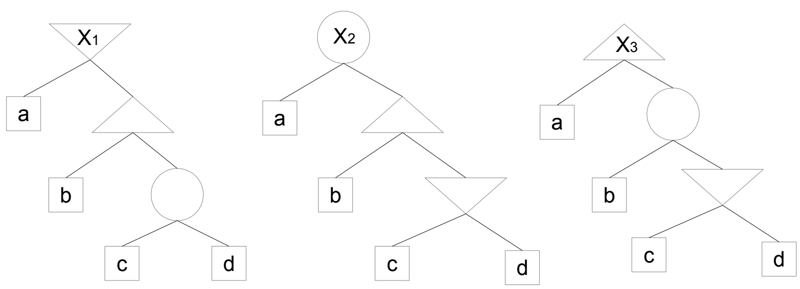


(d) Let X_1, X_2, and X_3 be the values at each root in the above minimax game trees. In these trees a, b, c, d, e, and f are constants (they are the same across all three trees). Determine which of the following statements are true for all possible assignments to constants a, b, c, d, d, and f.

(i) $X_1 = X_2$ True False

(ii) $X_1 = X_3$ True False

(iii) $X_2 = X_3$ True False



(e) In this question we want to determine relations between the values at the root of the new game trees above (that is, between X_1, X_2, and X_3).

All three game trees use the same values at the leaves, represented by a, b, c, and d. The chance nodes can have any distribution over actions, that is, they can choose right or left with any probability. The chance node distributions can also vary between the trees.

For each case below, write the relationship between the values using $<$, $\leq$, $>$, $\geq$, $=$, or NR. Write NR if no relation can be confirmed given the current information. Briefly justify each answer (one sentence at most). (Hint: try combinations of $\{-\infty, -1, 0, 1, +\infty\}$ for a, b, c, and d.)

(i) X_1 blank X_3

(ii) X_2 blank X_3