Exercise 1

Draw the Binary Decision Diagram associated with the following functions $f : \mathcal{B}^4 \rightarrow \mathcal{B}$ (having all 4 variables as input) under variable ordering x_1, x_2, x_3, x_4. For each BDD, indicate how many nodes and how many satisfying assignments represents.

1. $x_1 \vee x_3$
2. $\neg x_4$
3. $x_4 \wedge \neg x_2 \wedge \neg x_1$
4. $(x_1 \vee x_3) \wedge (x_2 \vee x_4)$

Exercise 2

For each of the BDDs in Exercise 1, does the variable ordering chosen minimize the number of nodes? Whenever a better variable ordering exist, provide it and draw the resulting BDD.

Exercise 3

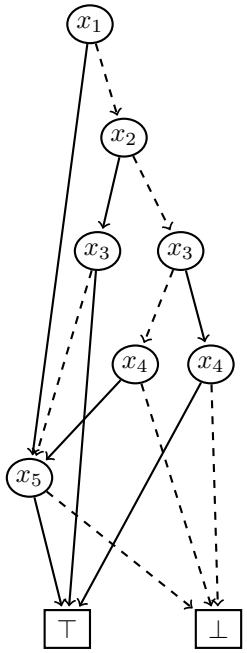
Consider the following statements. Indicate whether they are true or false. In case they are false, indicate why.

Note: we denote by $|f|$ the number of nodes used to represent a Boolean function f.

- For any Boolean function there is always a single, unique, BDD that represents it.
- Each path from the root of the BDD to a leaf node corresponds exactly with a total assignment of all variables.
- If a Boolean function f is represented as a BDD, then for each satisfiable assignment there is a corresponding path from the root node to $\top$.
- Consider 3 Boolean functions f,g, and h such that $h = f \vee g$. All of them are represented as BDDs under the same variable ordering. Then, $|f| + |g| \leq |h|$.
- Consider 3 Boolean functions f,g, and h such that $h = f \vee g$. All of them are represented as BDDs under the same variable ordering. Then, $\max(|f|, |g|) \leq |h|$.

Exercise 4

Consider the following BDD. What is the number of satisfying assignments of the Boolean function that it represents? To count it, run the model counting algorithm, labelling each node with the corresponding $mc(f)$ value.



Exercise 5

Compute the conjunction of these two BDDs, and draw the resulting BDD. Mark each node of the resulting BDD with identifiers (either re-using one of the existing identifiers or choosing a new one: $C_1, \ldots,$). For each recursive call to the *apply* procedure, specify what is the input (the identifiers of the two input nodes) and the output (the identifier of the output node).