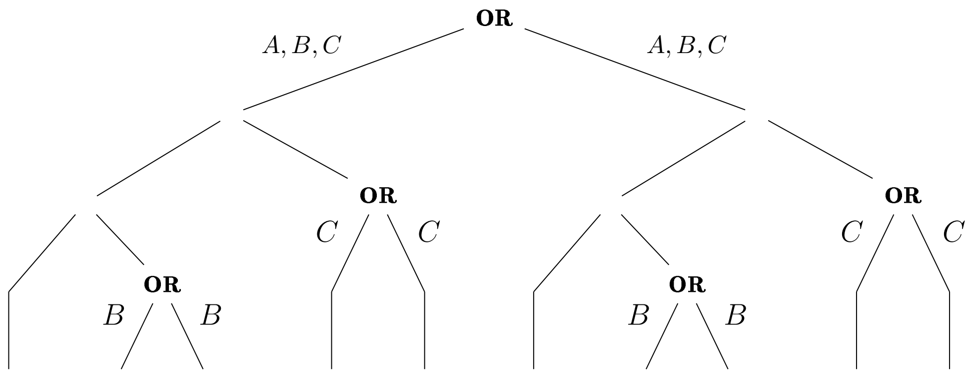
# Exercise Session – Solutions

## 1 Properties of probabilistic logic circuits

1. (a) Because the leaves are distinct, it's easy to verify that leaves with only one parent are not shared by a conjunction. For leaves with multiple parents, we follow them upwards and check if they don't share a common AND ancestor. As this is not the case, it's **decomposable**.

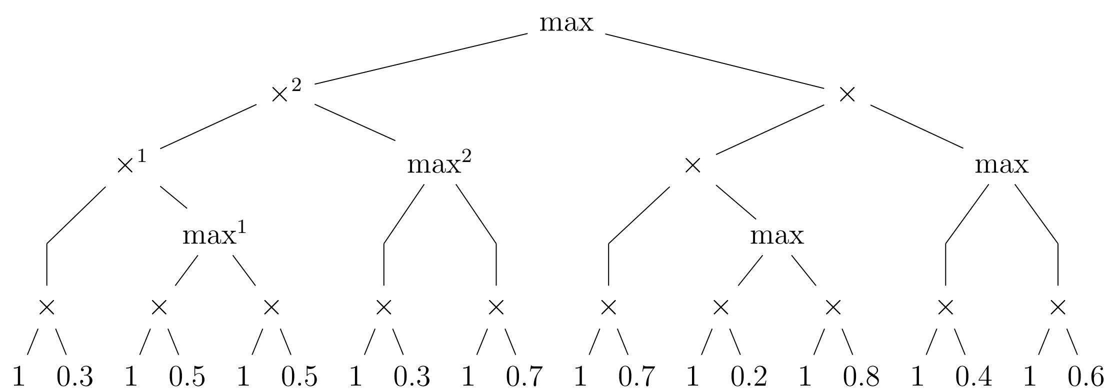
   (b) We can see that all disjuncts contains two conjuncts – one with a variable and one with its negation, e.g. $(B \wedge \theta_{B|\neg A}) \vee (\neg B \wedge \theta_{\neg B|\neg A})$. These can never be both satisfied and thus the circuit is **deterministic**.

   (c) For each of the OR statements we check the sets of variables of its operands, not considering the $\theta$.




   As they seem to be equal everywhere, the circuit is **smooth**.

2. We change AND in multiplications, OR in max and set the weights accordingly. As we yet have to make a decision about all variables, all of the $V$ and $\neg V$ weights are 1.



Every max makes a choice about an assignment for a variable. For example, $\max^1$ decides on $B = 1$ or $B = 0$, which is undecided as both have a weight of 0.5; $\max^2$ decides on $C = 1$ or $C = 0$, for which the latter has the highest weight (0.7).

The $\times$ compute the joint weight for these assignments. For example, $\times^1$ computes the joint weight of $A = 1$ and whatever choice for $B$ was made; $\times^2$ computes the weight for a complete assignment $(A = 1, C = 0, B = \{0, 1\})$, which is

$$(0.3 \times \max(0.5, 0.5)) \times \max(0.3, 0.7) = 0.105.$$

Similarly, the right branch has a weight of

$$(0.7 \times \max(0.2, 0.8)) \times \max(0.4, 0.6) = 0.336$$

for $(A = 0, B = 0, C = 0)$ which is greater than that of the left branch and is thus the MAP assignment.

# 2 Inference in a Bayesian network using WMC

For non-negated $\theta$ variables, the weights are equal to the entry in the conditional probability table. All other weights are 1. The theory that we need to be satisfied for $A = 1$ is

$$\begin{array}{llll} & (\theta_A \Leftrightarrow A) & \wedge & (\theta_{\neg A} \Leftrightarrow \neg A) \\ \wedge & (\theta_{B|A} \Leftrightarrow A \wedge B) & \wedge & (\theta_{\neg B|A} \Leftrightarrow A \wedge \neg B) \\ \wedge & (\theta_{B|\neg A} \Leftrightarrow \neg A \wedge B) & \wedge & (\theta_{\neg B|\neg A} \Leftrightarrow \neg A \wedge \neg B). \end{array}$$

We fill in both possible values for $B$ and then assign weights to the $\theta$ variables so that the theory is satisfied. For example, for $A$ and $B$ both equal to 1, we require that $\theta_A = 1$ for the first term in the conjunction and $\theta_{B|A} = 1$ for the third term to be true. All others should be 0 as the other side of the $\Leftrightarrow$ is 0 and $0 \Leftrightarrow 0$ is true.

| $A$ | $B$ | $\theta_A$ | $\theta_{\neg A}$ | $\theta_{B\|A}$ | $\theta_{\neg B\|A}$ | $\theta_{B\|\neg A}$ | $\theta_{\neg B\|\neg A}$ | $w$ |
|---|---|---|---|---|---|---|---|---|
| 1 | 1 | 1 | 0 | 1 | 0 | 0 | 0 | 0.1 |
| 1 | 0 | 1 | 0 | 0 | 1 | 0 | 0 | 0.4 |

The weights for these models is then computed by multiplying the weights for the variables that are 1. We can now easily answer the questions.

a) $p(A = 1)$ is the sum of the weights of all models where $A = 1$, or

$$p(A = 1) = 0.4 + 0.1 = 0.5$$

b) The most likely state for $B$ is the one where the model given $A = 1$ has the highest weight, or $B = 0$.

# 3 More inference in a Bayesian network using WMC

Similar to the previous exercise.

## 4 Model in ProbLog

```
0.7::burglary.
0.2::earthquake.

0.9::alarm :- burglary, earthquake.
0.8::alarm :- burglary, \+earthquake.
0.1::alarm :- \+burglary, earthquake.

evidence(alarm,true).

query(burglary).
query(earthquake).
```