

# COMPTE RENDU TME 1LRC :

Dans cette séance nous avons eu une première prise en main de LoTREC en vérifiant la validité/satisfiabilité des formules vues en TD. On a aussi pu mettre un problème concret sous une forme en formule de vérité. Grâce à ça on a pu voir la correspondance entre prémodelle et table de vérité.

## EXERCICE 1.2 :

### QUESTION 1 :

- F1 : Toutes les feuilles sont fermées la formule est donc insatisfiable

```
A & ~ (B -> A)
  A
  ~ (B -> A)
    B
    ~ A
    FALSE
```

- F2 : Il y a au moins une branche ouverte donc elle satisfiable. On s'intéresse donc à not(F2) qui est insatisfiable. F2 est donc valide.

```
(A v C) & (B v C) -> ~ B -> A & B v C
~ ((A v C) & (B v C)) v (~ B -> A & B v C)
~ ((A v C) & (B v C))
~ (A v C) v ~ (B v C)
~ (A v C)
~ A
~ C
```

- F3 : La formule logique est insatisfiable car toutes les branches sont fermées. On en conclut que not(F3) est valide.

prémodelle 1	prémodelle 2
<pre>~ (A -&gt; B -&gt; ~ B -&gt; ~ A)   A -&gt; B   ~ (~ B -&gt; ~ A)     ~ B     ~ (~ A)       A       ~ A v B         B         FALSE</pre>	<pre>~ (A -&gt; B -&gt; ~ B -&gt; ~ A)   A -&gt; B   ~ (~ B -&gt; ~ A)     ~ B     ~ (~ A)       A       ~ A v B         B         FALSE</pre>

- F4 : Il y a 6 branches ouvertes lors de l'exécution donc F4 est satisfiable.

```

(A -> B) & (B -> C) v (C -> B) & (B -> A)
(A -> B) & (B -> C)
A -> B
B -> C
~ A v B
~ B v C
~ A
~ B

```

- F5 : Une branche ouverte nous permet de dire que F5 est satisfiable. Comme  $\text{not}(F5)$  a aussi une branche ouverte alors elle est aussi satisfiable. On ne peut donc pas conclure sur la validité de la formule.

```

premodel 1
A -> B -> (B -> C <-> A -> C)
~ (A -> B) v (B -> C <-> A -> C)
~ (A -> B)
A
~ B

```

```

premodel 2.1
~ (A -> B -> (B -> C <-> A -> C))
A -> B
~ (B -> C <-> A -> C)
(B -> C) v (A -> C)
~ (B -> C) v ~ (A -> C)
~ A v B
A -> C
~ A v C
~ (B -> C)
~ A
B
~ C

```

- F6 : Ici toutes les branches sont ouvertes, F6 est donc satisfiable. On vérifie  $\text{not}(F6)$  qui est insatisfiable donc F6 est valide.

```

premodel 1
(A -> B) & (B -> C) -> A -> C
~ ((A -> B) & (B -> C)) v (A -> C)
~ ((A -> B) & (B -> C))
~ (A -> B) v ~ (B -> C)
~ (A -> B)
A
~ B

```

```

premodel 1
~ ((A -> B) & (B -> C) -> A -> C)
(A -> B) & (B -> C)
~ (A -> C)
A -> B
B -> C
A
~ C
~ A v B
~ B v C
B
C
FALSE

```

```

premodel 2.1
~ ((A -> B) & (B -> C) -> A -> C)
(A -> B) & (B -> C)
~ (A -> C)
A -> B
B -> C
A
~ C
~ A v B
~ B v C
B
C
FALSE

```

```

premodel 3
~ ((A -> B) & (B -> C) -> A -> C)
(A -> B) & (B -> C)
~ (A -> C)
A -> B
B -> C
A
~ C
~ A v B
~ B v C
B
C
FALSE

```

```

premodel 2.1
~ ((A -> B) & (B -> C) -> A -> C)
(A -> B) & (B -> C)
~ (A -> C)
A -> B
B -> C
A
~ C
~ A v B
~ B v C
B
C
FALSE

```

### EXERCICE 1.3 :

#### QUESTION 3 :

Code sans l'incertitude sur le fait que toux et fièvre implique grippe:

"and and and and Toux Sup imp Sup Fievre and imp and Fievre Toux Grippe imp Grippe Tamiflu not Tamiflu"

Code corrigé :

"and and and and Toux Sup imp Sup Fievre and imp and Fievre Toux or Bronchite Grippe imp Grippe Tamiflu not Tamiflu"

En ajoutant la bronchite il n'est pas indiqué de prendre du tamiflu dans tout les cas possible (car des certains pré-modèles sont valides par exemple).

