Logique Booleene

Definitions et props

Définition 1: Commutatif les variables peuvent etre inverses

Définition 2: L'arbre de Derivation C'est un format de pour representer une proposition

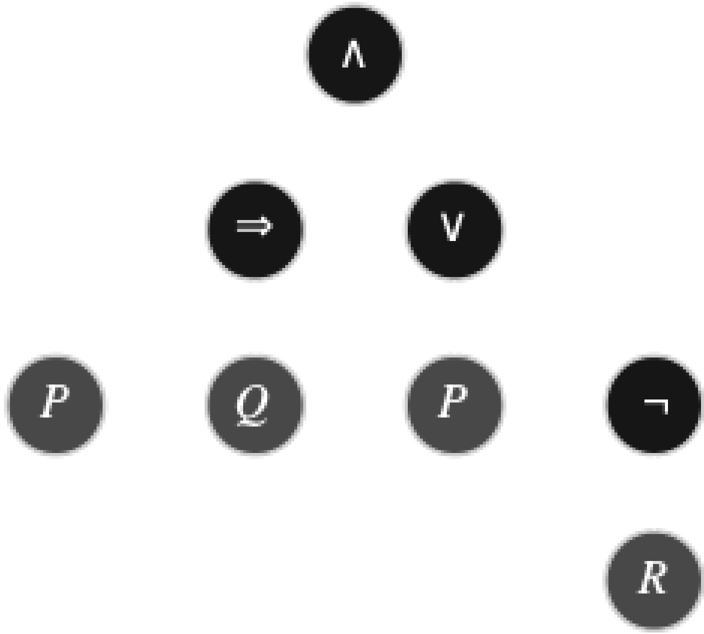

Figure 1: $(P \Rightarrow Q) \wedge (P \vee \neg R)$

Définition 3: Loi de De Morgan Soit P et Q deux assertions, alors
$\neg(P \vee Q) \equiv \neg P \wedge \neg Q$

$\neg(P \wedge Q) \equiv \neg P \vee \neg Q$

Tables de verite

il est assume qu'un connecteur est commutatif sauf mentione autrement

table de $\wedge$, dit conjonction, lu "et": q binaire

$\bot$	$\bot$	$\bot$
$\top$	$\bot$	$\bot$
$\bot$	$\top$	$\bot$
$\top$	$\top$	$\top$

table de $\vee$ dit disjonction, lu "ou": q binaire

$\bot$	$\bot$	$\bot$
$\bot$	$\top$	$\top$
$\top$	$\bot$	$\top$
$\top$	$\top$	$\top$

Définition 4: Clause on dit clause conjonctive (ou respectivement disjontive) toute formule composé de conjonction (respectivement disjonction)

table de $\oplus$: q binaire

$\bot$	$\bot$	$\bot$
$\bot$	$\top$	$\top$
$\top$	$\bot$	$\top$
$\top$	$\top$	$\bot$

table de $\Rightarrow$: q binaire dit non commutatif

$\bot$	$\bot$	$\top$
$\bot$	$\top$	$\top$
$\top$	$\bot$	$\bot$
$\top$	$\top$	$\top$

autrement dit, vrai sauf si p est vrai et q est faux

table de $\Leftrightarrow$: q binaire

$\bot$	$\bot$	$\top$
$\bot$	$\top$	$\bot$
$\top$	$\bot$	$\bot$
$\top$	$\top$	$\top$

vrai si les deux variables ont la meme valeur

Proprietes

- comutativite de $\wedge$ et $\vee$

$$(p \wedge q) \equiv (q \wedge p)$$

$$(p \vee q) \equiv (q \vee p)$$

- associativite de $\wedge$ et $\vee$

$((P \wedge Q) \wedge R) \equiv ((q \wedge R) \wedge P)$ $((P \vee Q) \vee R) \equiv ((Q \vee R) \vee P)$

- idempotence de $\wedge$ et $\vee$

$(p \wedge p) \equiv p$
$(p \vee p) \equiv p$

Modus{ponens, tollens}

Définition 5: modus ponen Soit P et Q deux propositions, si P est vrai et $P \Rightarrow Q$ est vrai, alors Q est vrai.

$$P, P \Rightarrow Q \dashv Q$$

car $P \Rightarrow q \equiv \neg Q \Rightarrow \neg P$

Définition 6: modus ponen Soit P et Q deux propositions, si $\neg Q$ est vrai et $P \Rightarrow Q$ est vrai, alors la proposition $\neg P$ est vrai.

$$\neg Q, P \Rightarrow Q \dashv \neg P$$

TPs

Question 1: Ecrire une fonction `interpretations(nbVar)` qui renvoie le tuple constitue de toutes les interpretations possible de nbvar variables propositionnelles

la technique que j'ai opte est de calculer tous les nombre possible en binaire jusqu'a 2^{nbvar}, puis de les retranscrire en tuple de vrai/faux. Voici le code (on assume une fonction translate to tuple defini comme le suit)

```
# Q1: ecrire une fonction Inter(nbvar) qui renvoie le tuple constitue de toutes les interpretations possible de nbvar variables propositionnelles
def translatetotuple(binary: str):
    result = []
    for i in binary:
        if i == '1':
            result.append(True)
        else:
            result.append(False)

    return tuple(result)

def inter(nbvar):
    finalresult = ()
    for i in range(nbvar**2):
        result = bin(i)
        result = result[2:]
        while len(result)<nbvar:
            result = '0'+result
        result = translatetotuple(result)
        finalresult += result,
    return finalresult
```