NOTE : $/20$

INF202

EXAM BLANC
2021

À lire attentivement avant de commencer le sujet :

— Justifier proprement vos réponses ; vous ne recevrez pas tous les points pour une réponse correcte sans justification. Vous pouvez énoncer des résultats du cours sans les démontrer.
— Aucun document n'est autorisé à l'exception d'une feuille A4 recto-verso.
— Les appareils électroniques sont interdits.
— Vous devez répondre sur le sujet.
— Vous ne devez pas répondre au crayon à papier.

Exercice 1 : Modèles de propositions (4 points)

Toutes les propositions logiques suivantes sont satisfiables. Donner un modèle pour chacune.

1. $(a \vee b) \wedge \neg a$
2. $\neg((a \wedge b) \Rightarrow (\neg b \vee c))$
3. $(\bot \wedge a) \vee \neg(a \wedge \top)$
4. $(a \vee b \vee c) \wedge (\neg a \vee \neg b \vee \neg c) \wedge (a \vee \neg b \vee \bot) \wedge \top \wedge \neg c$

Exercice 2 : Représentation de relations (3 points)

Considérons les relations suivantes :

— $R = \{(a_1, b_1), (a_2, b_3), (a_3, b_1), (a_3, b_2)\}$ définie sur $\{a_1, a_2, a_3\} \times \{b_1, b_2, b_3\}$
— $S = \{(c_1, b_1), (c_1, b_2), (c_2, b_1), (c_3, b_3), (c_4, b_1), (c_4, b_3)\}$ définie sur $\{c_1, c_2, c_3, c_4\} \times \{b_1, b_2, b_3\}$
— $T = \{(a_1, c_1), (a_1, c_2), (a_2, c_2), (a_3, c_3), (a_3, c_4)\}$ définie sur $\{a_1, a_2, a_3\} \times \{c_1, c_2, c_3, c_4\}$

Représenter chacune de ces relations sous la forme d'une matrice

Exercice 3 : Equivalence de propositions (4 points)

Les propositions suivantes sont forment des paires équivalentes. Réunir ces paires en justifiants vos réponses.

$\top \qquad \bot \qquad \neg a \Rightarrow (a \vee \neg b) \qquad (\neg a \wedge b) \Rightarrow a \qquad (a \vee \neg a) \Rightarrow (\neg b \wedge b) \qquad (a \Rightarrow b) \vee (a \wedge \neg b)$

Exercice 4 : Arbres et sous-formules (4 points)

Pour la proposition suivante, construire son arbre et donné chacune de ses sous-formules sous forme infixe et préfixe :

$$\neg(\neg a \Rightarrow \neg(b \vee \neg c)) \Leftrightarrow (c \Rightarrow \neg a)$$

Exercice 5 : Algorithmes récursifs sur les formules logiques (5 points)

Cet exercice a vocation à tester vos connaissances algorithmiques. Aussi, de légères erreurs de syntaxe en python ne seront pas pénalisées si l'algorithme que vous décrivez est correcte et une méconnaissance du noms des fonctions données en TP ne sera pas pénalisée tant que vous explicitez ce que font vos fonctions.

1. Compléter le code python suivant, permettant d'obtenir la fonction `comptImpl` qui prend en entrée une formule et renvoie le nombre de fois que l'implication apparaît dans la formule.

```
def compteImpl(f):
    n = f.nb_operandes()
    if n == 0:
            return ......
    elif n == 1:
        f2 = (f.decompose())[0]
        return .......
    else:
        [f1,f2]= f.decompose()
        op = f.operateur()
        if op == .......  :
            return ......
        else :
            return ......
```

2. Écrire en python la fonction `soustractionpq` qui prend en entrée une formule et deux noms de variables et qui renvoie le nombre d'apparition de la première variable moins le nombre d'apparition de la deuxième variable.

   (/!\ on ne souhaite pas compter le nombre de fois qu'apparaît la première variable puis le nombre de fois qu'apparaît la seconde variable et soustraire ces deux valeurs, tout dois être fait dans une seule fonction récursive)

3. Écrire en python la fonction `simplifie` qui prend en entrée une formule et renvoie une formule simplifié en prenant en compte les simplification suivantes :
   — $\top \wedge p \equiv p$
   — $\top \vee p \equiv \top$
   — $\bot \wedge p \equiv \bot$
   — $\bot \vee p \equiv p$
   — $p \wedge \neg p \equiv \bot$
   — $p \vee \neg p \equiv \top$
   Ainsi $(p \wedge \neg p) \vee q \equiv \bot \vee q \equiv q$ et sur l'entrée $(p \wedge \neg p) \vee q$ votre fonction devra donc renvoyer q.