L3 Info

CONTROLE FINAL DE PROGRAMMATION LOGIQUE ET FONCTIONNELLE

LOGIQUE PROPOSITIONNELLE

EXERCICE 1 (3 POINTS)

Voici des formules. Donnez la liste de celles qui sont cohérentes et la liste de celles qui sont valides (i.e., qui sont des tautologies).

1. $(a \vee b) \wedge (\neg a \vee b) \wedge (a \vee \neg b)$
2. $a \rightarrow \neg a$
3. $(a \rightarrow b) \wedge (a \rightarrow \neg b)$
4. $(a \rightarrow b) \vee (a \rightarrow \neg b)$
5. $a \rightarrow (b \vee \neg b)$
6. $(b \wedge \neg b) \rightarrow a$

EXERCICE 2 (1 POINT)

Donnez un modèle de la formule suivante :

$(\neg a \vee \neg b \vee \neg c) \wedge (\neg a \vee b \vee \neg d) \wedge (a \vee \neg b \vee \neg e) \wedge (a \vee b \vee \neg f) \wedge c \wedge e \wedge f$

EXERCICE 3 (2 POINTS)

On dit que la suite Booléenne $x_1, x_2, \ldots, x_n$ est un bargraph si elle est constituée d'une séquence éventuellement vide de 1 suivie uniquement de 0. Par exemple, les bargraphs de 4 valeurs sont 0000, 1000, 1100, 1110 et 1111.

Donnez une formule propositionnelle qui modélise la propriété : "les valeurs des variables v_1, v_2, v_3, v_4 représentent un bargraph."

LOGIQUE DU 1ER ORDRE

EXERCICE 4 (2 POINTS)

Soient les deux formules suivantes :

- $\Sigma_1 = \forall X \forall Y \forall Z [(r(X,Y) \wedge r(Y,Z)) \rightarrow r(X,Z)]$
- $\Sigma_2 = \forall X \forall Y [r(X,Y) \rightarrow r(Y,X)]$

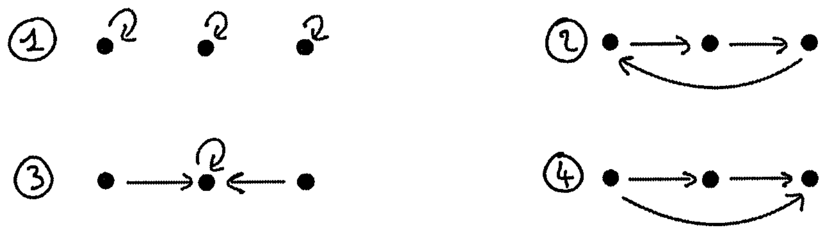
Donnez une interprétation I_1 qui satisfait Σ_1 et falsifie Σ_2. Donnez une interprétation I_2 qui falsifie Σ_1 et satisfait Σ_2. Les deux interprétations doivent avoir le domaine $\{\alpha, \beta, \omega\}$.

EXERCICE 5 (3 POINTS)

Voici deux formules

- $\Omega_1 = \forall X \exists Y p(X,Y)$
- $\Omega_2 = \exists Y \forall X p(X,Y)$

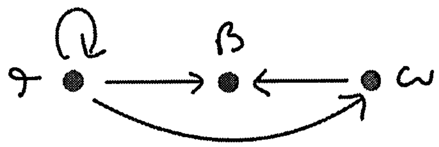
et 4 interprétations du prédicat p numérotée de 1 à 4.



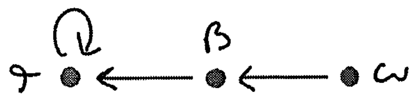
Donnez la liste des interprétations qui, parmi les 4 proposées, satisfont Ω_1. Donnez la liste des interprétations qui, parmi les 4 proposées, satisfont Ω_2.

EXERCICE 6 (1 POINT)

On considère une relation r sur un domaine d'interprétation D. On dit que r relie X à Y si et seulement si le couple (X,Y) appartient à r. Dans l'exemple ci-dessous, r relie α à α, α à β, α à ω et ω à β.



On appelle relation **fonctionnelle** sur $D \times D$ toute relation qui relie tout élément de D à au plus un élément de D (Sur le graphique, de chaque élément part soit aucune flèche, soit une seule flèche). Par exemple la relation présentée en exemple ci-dessus n'est **pas** fonctionnelle car elle relie α à 3 éléments. Par contre, la relation représentée ci-dessous **est** fonctionnelle.



Par convention, on considère que le prédicat egal/2 s'interprète comme l'égalité standard, i.e., egal(X,Y) est vrai si et seulement si X et Y ont la même valeur, i.e. représentent le même élément du domaine d'interprétation.

Donnez une formule de la logique du premier ordre (utilisant les prédicats egal/2 et $r/2$) qui modélise la propriété : "la relation r est fonctionnelle".

PROLOG

EXERCICE 7 (2 POINTS)

Soit le programme suivant.

```
val(1).
val(2).
val(3).

gen([X,Y]) :- val(X), val(Y).
```

Le but gen(L) produit toutes les listes de deux valeurs prises parmi 1,2,3 : [1,1], [1,2], [1,3], [2,3], etc.

Définissez un prédicat genere/2 tel que le but genere(N,L), avec N un entier en entrée, fasse remonter dans la variable de sortie L toutes les listes de N éléments parmi ceux pouvant satisfaire le prédicat val/1. Par exemple, le but genere(3,R) fait remonter dans R toutes les listes de trois éléments parmi 1,2,3. Le but genere(2,L) produit les mêmes listes que le but gen(L).

```
genere(0, --------- ).

genere(N, [T|Q]) :- N>0, N1 is N-1, ------------------------------------------ .
```

EXERCICE 8 (2 POINTS)

Définissez un prédicat ins/3 tel que ins(L,X,R), où L est une liste d'entrée et X une valeur d'entrée, fasse remonter dans la variable de sortie R toutes les listes pouvant être obtenues en insérant X à une position quelconque dans L. Par exemple, le but ins([1,2],3,R) a pour résultat R=[3,1,2], R=[1,3,2], R=[1,2,3].

Indice : Pour produire la liste de sortie, il faut soit ajouter X au début de la liste d'entrée, soit l'insérer dans la queue de la liste d'entrée en conservant son élément de tête.

EXERCICE 9 (2 POINTS)

On représente des expressions arithmétiques par des termes fonctionnels. Les entiers 0, 1, 2... sont représentés par les termes fonctionnels v(1), v(2), v(3)... Une somme de deux expressions arithmétiques A est B est représentée par plus(A,B), un produit de deux expressions arithmétiques A et B est représentée par mul(A,B). Par exemple, l'expression $(2 \times 3) + (7 \times 5)$ sera représentée par le terme fonctionnel :

```
plus( mul(v(2),v(3)), mul(v(7),v(5)) )
```

Vous devez définir le prédicat eval/2 tel que le but eval(E,R) fasse remonter dans la variable de sortie R la valeur de l'expression arithmétique représentée par le terme fonctionnel E donné en entrée. Par exemple, le but suivant...

```
eval(plus(mul(v(2),v(3)), mul(v(7),v(5))), R).
```

... produit le résultat R = 41.

EXERCICE 10 (2 POINTS)

Complétez la définition du prédicat stat/3 qui permet de compter le nombre d'additions et multiplications dans une expression arithmétique. Par exemple, le but...

```
stat(plus(mul(v(2),v(3)), mul(v(7),v(5))), A, M).
```

... produit le résultat A = 1, M = 2 car l'expression donnée en entrée dans le premier slot comporte 2 multiplications et une addition.

```
stat(v(_),0,0).
stat(plus(X,Y),A,M) :- stat(X,A1,M1), stat(Y,A2,M2), A is --------, M is -------------.
stat(mul(X,Y),A,M) :-  ---------------------------------------------------------------.
```