Université d'Orléans - M1 Informatique - Module PLC - TD $n^o 2$
2012-2013

Préliminaires

- Le symbole "=" en prolog signifie l'unification terme à terme et pas l'affectation. Pour unifier une valeur ou expression numérique avec une variable, en évaluant mathématiquement le résultat, il faut utiliser "is". Par exemple :

```
?- M=1+1, N is 1+1.
M = 1+1
N = 2 ;
No
?- N is 1+1, M is N+3, P = N+M.
N = 2
M = 5
P = 2+5 ;
No
```

- Il existe quelques actions prédéfinies comme write(), qui est applicable à une constante, une variable ou une chaîne de caractères entre guillemets simples, nl (passage à la ligne). Par exemple :

```
?- write('bonjour'),nl,write('toi').
bonjour
toi
Yes
?- X is 2, write(X).
2
X = 2 ;
No
```

- En prolog, on ne définit pas de fonction ou de procédures mais des prédicats (à valeur dans 0 , 1), éventuellement avec 0 argument. Donc :
  - Pour faire l'équivalent d'une procédure, on dit qu'un certain prédicat est vrai à condition qu'un certain nombre d'actions soient faites, par exemple :
    `affiche :- write('bonjour').`
  - Pour définir une fonction $f(X_1, ..., X_n) = Y$, on definit un prédicat $p(X_1, ..., X_n, Y)$ qui est vrai si $f(X_1, ..., X_n) = Y$, par exemple :
    `a_pour_successeur(X,Y) :- Y is X+1.`

1. Comment exprimer qu'on veut calculer la somme de deux nombres ?
2. Comment exprimer qu'on veut connaître le plus grand parmi 2 nombres ? parmi 3 ?
   Que produit max2(X,1,3) ? Pourquoi ? Même question avec max2(1,X,3).

Exercices sur la récursivité

En prolog, il n'y a pas d'autres moyens de faire des boucles que d'utiliser la récursivité.

1. Afficher N fois 'bonjour'. (Ou : ecrit(N) est vrai si le message 'bonjour' est écrit N fois.)
2. Afficher les nombres de 1 à N.
   On pourrait penser que la virgule, qui est la traduction en prolog du ET logique, devrait être commutative. Or il est évident dans cet exemple que ce n'est pas le cas (sinon les deux prédicats seraient identiques). Ceci est dû à ce qu'on appelle des effets de bord : si on reste dans le domaine de l'évaluation d'expressions logiques, c'est bien commutatif, mais l'ordre d'évaluation sera différent, et comme ici on produit des affichages en cours d'évaluation, le résultat obtenu est différent.
3. Dire si un nombre est pair.
4. Trouver la somme des N premiers entiers. (Ou : som(N,X) est vrai si X est la somme des entiers de 1 à N.)
5. Trouver la factorielle d'un nombre. (Ou : fact(N,X) est vrai si X vaut N!.)
6. fibo(N,X) est vrai si X est la valeur de la suite de Fibonacci au rang N.
7. coef(N,P,C) a un sens si $N > P$ et est vrai si C vaut N!/((N-P)!P!) ; utiliser la relation de Pascal.
   On demande de calculer les C(n,p), coefficients du binôme :

   $(a+b)^n = C(n,0)a^n b^0 + C(n,1)a^{n-1}b^1 + ... + C(n,p)a^{n-p}b^p + ... + C(n,n-1)a^1 b^{n-1} + C(n,n)a^0 b^n$

   où $C(n,p) = n!/(n! \times (n-p)!)$.

   Ce qui donne par exemple : $(a+b)^3 = a^3 + 3a^2b + 3ab^2 + b^3$.
   La relation de Pascal nous dit que $C(n,p) = C(n-1,p) + C(n-1,p-1)$. Ce qui permet de représenter les $C(n,p)$ ainsi :

```
        p=0   p=1   p=2   p=3   p=4   p=5   p=6   p=7   p=8
n=0     1     |     |     |     |     |     |     |     |
n=1     1     1     |     |     |     |     |     |     |
n=2     1     2     1     |     |     |     |     |     |
n=3     1     3     3     1     |     |     |     |     |
n=4     1     4     6     4     1     |     |     |     |
n=5     1     5    10    10     5     1     |     |     |
n=6     1     6    15    20    15     6     1     |     |
n=7     1   7 +  21    35 + 35    21     7     1     |
                   ||          ||
n=8     1     8    28    56    70    56    28     8     1
```

   La relation de Pascal énonce que chaque case est la somme de celle immédiatement au-dessus et de celle au-dessus à gauche. Par exemple le 70 de la dernière ligne est la somme des deux 35 de la ligne du dessus. (Vous pouvez vérifier, ça marche partout !)