Algorithmique-Programmation : IAP1 - contrôle continu - CORRIGÉ
Lundi 6 octobre 2008 - Sans documents - durée : 30 minutes
Répondez directement sur la feuille.

Nom : ..... **Prénom : .....** **Groupe : .....**

Exercice 1

Compléter la session suivante. Il n'y pas d'erreur syntaxique mais des erreurs de typage peuvent exister. Dans ce cas, indiquez la réponse "erreur de typage" et expliquez en quelques mots pourquoi. On rappelle qu'une réponse de Ocaml est de la forme `val idf : type = valeur` ou `- : type = valeur`.
Remarque : aucune réponse n'est à donner pour la déclaration des types `t` et `tt`.

```
#let f b = (b + 2, b<10) ;;
val f : int -> int * bool = <fun>

#f 3;;
- : int * bool = (5, true)

#let g x = if x mod 2 = 0 then  x+1 else x/2;;
val g : int -> int = <fun>

#(g 5, g 4);;
- : int * int = (2, 5)

#let h (x,y) = (y, x+2);;
val h : int * 'a -> 'a * int = <fun>

#let j (x,y) =  [x;x];;
val j : 'a * 'b -> 'a list = <fun>

#let b = h (0,5) in let c = h b in c;;
- : int * int = (2, 7)

#b;;
Unbound value b

#let quid l = match l with
    [] -> 1
  | [e] -> if e then 2 else 3
  | [e1;e2] -> if e1 then 4 else (if e2 then 5 else 6)
  | _ -> 7;;
val quid : bool list -> int = <fun>

#quid [true] ;;
- : int = 2

#quid [true;true] ou [true;false] ;;
- : int = 4
```

```
#quid [false;false] ;;
- : int = 6

#quid [true;false; false; true]  différentes réponses possibles ici ;;
- : int = 7

#let toto (l,x) = match l with
| (a,b)::r -> a = (fst x)
| _ -> true;;
val toto : ('a * 'b) list * ('a * 'c) -> bool = <fun>,

type t = T of bool*string | P of  string;;

# [ T (true, "toto") ; P "titi"] ;;
- : tt list = [ T (true, "toto") ; P "titi"]

type 'a tt = T of bool*'a | P of 'a ;;

#let valeur = let a = 4 in P a;;
 valeur : int tt = P 4

#let  m (x, y) = match x with
  T (_,b) ->  y=b
| P (a) ->  y=a ;;

val m : 'a tt * 'a -> bool = <fun>

#m (valeur, 4);;
 - : bool = true

#m (valeur, "bc");;
Characters 2-16:
  m (valeur, "bc");;
    ^^^^^^^^^^^^^^
This expression has type int tt * string but is here used with type
  int tt * int
```

Exercice 2

Écrire la fonction `garder` dont l'interface est donnée ci-dessous.

```
(*interface garder
type : 'a list*int -> 'a list
args l, n
precondition : true
postcondition : retourne la liste contenant les n premiers élements de l
                (si n > longueur l, on retourne les éléments de l)
raises : rien
tests garder ([1;2;3;4], 3) (*[1;2;3]*),
      garder ([1;2;3;4], 5) (*[1;2;3;4]*),
      garder ([1;2;3;4], 0) (*[]*),
*)

let rec garder (n,l) = match l with
[] -> []
| e::r -> if n=0 then [] else e::garder(n-1,r);;
```

Exercice 3

Une date est soit la donnée de 3 entiers : un numéro de jour, un numéro de mois et un numéro d'année soit la donnée d'un numéro de jour et de l'année (dans ce cas on exprime que c'est le nième jour de l'année).

Définir à l'aide d'un type somme le type des dates.

```
type date = Triplet of int*int*int | Nieme of int*int;;
```

Déclarer la date correspondant au 1er février 2008 en utilisant les deux formats possibles.

```
let d1 = Triplet (1, 2, 2008);;
- : date = Triplet (1, 2, 2008)

let d2 = Nieme (32, 2008);;
- : date = Nieme (32, 2008)
```

Écrire la fonction `saint_sylvestre` de type `date -> bool`, qui teste si le la date passée en argument correspond à un jour de la Saint Sylvestre (dernier jour de l'année). Pour tester si une année est bissextile on pourra utiliser la fonction `bissextile` de type `int -> bool` qui retourne `true` si l'année paramètre est bissextile et `false` sinon (n'écrivez pas la fonction `bissextile`).

```
let saint_sylvestre jour = match jour with
    Triplet (x, y, _) -> x=31 && y=12
  | Nieme (x,_) -> if bissextile y then x=365 else x=366;;
```

Exercice 4 (Commandes Unix)

1. Quelle différence y a-t-il entre les commandes `mv toto titi` et `cp toto titi` ?

cp (copy) fait une copie du premier fichier : un nouveau fichier est créé, son contenu est identique à celui du fichier d'origine, mais leurs noms sont différents. toto et titi existent de façon parallèle. mv (move) renomme le fichier toto en titi. Apres l'exécution de la commande, toto n'existe plus et titi a le contenu de toto (avant le mv).

2. Je veux aller dans le répertoire `/usr/local/games/mariobros`, et le répertoire courant est `/usr/local`. Quelle(s) commande(s) peut-on taper ?
   A : `cd /games/mariobros`
   B : `cd games/mariobros`
   C : `cd local/mariobros`
   D : `cd /usr/local/games/mariobros`
   E : `cd /usr/local/../local/games/mariobros`
   F : `cd ../games/mariobros`

   B - D - E

3. Supposons

   ```
   $ cat devinette.txt
   devinette numero 4:
   pince mi et pince moi
   sont dans un bateau.
   pince mi tombe 'a l eau.
   qui est ce qui reste ?
   ```

   Qu'affiche la commande suivante : `cat devinette.txt | grep ce | wc -l` ?

   A : 0 B : 1 C : 2 D : 3 E : 4 F : 5

   D

   Qu'affiche la commande suivante : `cat devinette.txt | grep 4 | wc -l` ?

   A : 0 B : 1 C : 2 D : 3 E : 4 F : 5

   B