UNIVERSIDAD DE MÁLAGA
E.T.S.I. INFORMÁTICA

Programación Declarativa

(3º de Ingeniería Técnica en Informática)
17 de Febrero de 2005

Alumno: _____ Grupo: _____

Prolog

Ejercicio 1

$(a)(2\ pts.)$ Realiza el árbol de búsqueda para el objetivo `p([a,b,c,b,a],X)` y el programa *Prolog*

```
p([X],X).
p([X|Xs],Y) :-
        concatena(As,[X],Xs),
        p(As,Y).
concatena([],Ys,Ys).
concatena([X|Xs],Ys,[X|Zs]):-
        concatena(Xs,Ys,Zs).
```

$(b)(1\ pto.)$ Completa la tabla de comportamiento del predicado `p` anterior.

$(+,+)$		
$(+,-)$		
$(-,+)$		
$(-,-)$		

Ejercicio 2

$(a)(1{,}25\ pts.)$ Define el predicado `genNat/1` que genera los números naturales:

```
?- genNat(N).
N = 0;
N = 1;
N = 2;
...
```

$(b)(1{,}25\ pts.)$ Define el predicado `genEnt/1` que genera los números enteros (naturales con signo):

```
?- genEnt(N).
N = 0;
N = 1;
N = -1;
N = 2;
...
```

Ejercicio 3

$(1{,}5\ pts.)$ Define el predicado `posiciones(AsXsBs,Xs,N)` tal que devuelva por revaluación en `N` las posiciones en las que aparece la sublista `Xs` en la lista `AsXsBs`.

```
?- posiciones([a,b,a,b,a,b,c,a,b,a],[a,b,a],N).
N  = 0;
N  = 2;
N  = 7;
No
```

Ejercicio 4

$(a)(1{,}5\ pts.)$ Define el predicado `cuenta(XsYs,X,Ys,N)` tal que dadas la lista `XsYs` y un término `X`, devuelva en `N` el número de `X` que aparecen al principio de `XsYs`, y en `Ys` la lista que resulta de eliminar todas las `X` por las que empieza `XsYs`

```
?- cuenta([5,5,5,4,4,1,6],5,N,Ys).
N = [4,4,1,6],
Ys = 3  ;
No
?- cuenta([5,5,5,4,4,1,6],1,N,Ys).
N = [5,5,5,4,4,1,6],
Ys = 0  ;
No
```

$(b)(1{,}5\ pts.)$ Define el predicado `nombra(Xs,Ys)` que dada una lista de enteros positivos `Xs`, devuelve una lista `Ys` que "nombra" la lista `Xs`, mencionando cuantas veces consecutivas aparece cada elemento de `Xs`

```
?- nombra([1,1,1,3,3,7,4,4,3],Ys).
Ys = [3,1,2,3,1,7,2,4,1,3] ;
No
```

UNIVERSIDAD DE MÁLAGA
E.T.S.I. INFORMÁTICA

Programación Declarativa

(3º de Ingeniería Técnica en Informática)
17 de Febrero de 2005

Alumno: _____ Grupo: _____

Haskell

Ejercicio 1

$(a)(0{,}5\ pts.)$ ¿ Cuál es el tipo polimórfico de la función `h` ?

```
h f g (x,y) = (f x, g y)
```

```
h :: _____
```

$(b)(0{,}5\ pts.)$ ¿Cuál es el tipo polimórfico de la función `h'` ?

```
h' f g (x,y) = (g (f x), f (g y))
```

```
h' :: _____
```

$(c)(0{,}5\ pts.)$ ¿Cuál es el tipo polimórfico de la función `f` ?

```
f g p h = map g . filter p . map h
```

```
f :: _____
```

$(d)(0{,}75\ pts.)$ Define la función `f` del apartado anterior usando una lista por comprensión

```
f g p h = _____
```

$(e)(0{,}75\ pts.)$ Define la función `f` del apartado anterior usando `foldr`

```
f g p h = foldr_____
            where
                _____
                _____
                _____
                _____
```

Ejercicio 2

$(a)(1\ pto.)$ Completa la definición de la función `span` que se utiliza para dividir una lista en dos de manera que la primera contiene todos los elementos consecutivos de la lista que verifican un predicado dado y la segunda el resto. Por ejemplo:

```
span (<6) [4,3,5,2,7,3,2,5,8,9] ==> ([4,3,5,2],[7,3,2,5,8,9])

span  :: (a -> Bool) -> [a] -> ([a],[a])
span p []        = ([],[])
span p (x:xs)
     | p x       =_____________
     | otherwise = ([],x:xs)
                       where (ys,zs) = span p xs
```

$(a)(1{,}5\ pts.)$ Un esquema simple de comprensión para cadenas de texto que contienen series de caracteres repetidos consiste en reemplazar cada una de las series por una única copia del carácter repetido y un dígito que indique el número de repeticiones. Por ejemplo, la cadena *aaaaabbbbcc* sería comprimida a *a5b4c2*. Como cada carácter va seguido de un único dígito, secuencias de más de 9 caracteres deben ser fraccionadas en bloques de no más de 9 caracteres. Por ejemplo, la cadena *aaaaaaaaaaa* podría comprimirse como *a9a3*.

Utilizando la función span anterior, completa las funciones `bloque` y `comprime` . La primera toma un carácter y un número y genera el bloque para ese carácter; la segunda toma una cadena de caracteres y la devuelva comprimida. Por ejemplo:

```
bloque 'a' 12                    ==> "a9a3"
comprime "aaaaabbbbbbbbbbbbcc"   ==> "a5b9b3c2"

bloque x n
  | n <= 9    = x:show n
  | otherwise =____________________

comprime [] = ""
comprime (x:xs) =_____________________________________________
           where (cx,rs) = span_______ xs
```

$(b)(1{,}5\ pts.)$ Escribe una función `descomprime` que tome una cadena de caracteres comprimida y devuelva la original. Por ejemplo:

```
descomprime  "a5b9b3c2"  ==> "aaaaabbbbbbbbbbbbcc"
```

Utiliza la función predefinida `read` que transforma una cadena en un entero:

```
read "7" ==>  7


expande []       = ""
expande (x:n:xs) =_____________________________________________
```

Ejercicio 3

Considera la siguiente definición de árboles con nodos de uno y dos hijos, no vacíos, y con información en nodos y hojas:

```
data ArbolI a = Hoja a | Uno  a (ArbolI a) | Dos (ArbolI a) a (ArbolI a)
              deriving Show
```

Se pretende utilizar este árbol para alojar elementos de manera que simule un árbol binario de búsqueda. En ese sentido, en un nodo `Dos`, la raíz debe ser menor o igual que los elementos de la rama izquierda y mayor que los de la derecha, mientras que en un nodo `Uno`, la raíz debe ser mayor o igual que los elementos de la rama inferior.

$(a)(0{,}75\ pts.)$ Define la función `inserta` que inserta un elemento en un árbol ordenado de este tipo de manera que se mantenga ordenado.

```
inserta ::_____________________________________________
inserta x (Hoja y)
  | x <= y = Uno_____________________
  | x > y  = Uno_____________________

inserta x (Uno y r)
  | x <= y = Uno_____________________
  | x > y  = Dos_____________________

inserta x (Dos ri y rd)
  | x <= y = Dos_____________________
  | x > y  = Dos_____________________
```

$(b)(0{,}75\ pts.)$ Dada la función

```
aArbol (x:xs) = aArbol' (Hoja x) xs
```

define la función `aArbol'` que inserta todos los elementos de la lista `xs` en un árbol ordenado manteniéndolo ordenado.

```
aArbol' =_____________________________________________________________
```

$(c)(0{,}5\ pts.)$ Define la función `enOrden` que hace un recorrido en orden de los elementos de manera que la lista resultante estará ordenada.

```
enOrden :: _____________________________
enOrden (Hoja x) =_____________
enOrden (Uno x r) =_____________________
enOrden (Dos ri x rd) =_________________
```

$(d)(1\ pto.)$ Si consideramos la siguiente función de plegado para estos árboles

```
foldI :: (b -> a -> b -> b) -> (a -> b -> b) -> (a -> b) -> ArbolI a -> b
foldI f g h (H x)         = h x
foldI f g h (Uno x r)     = g x (foldI f g h r)
foldI f g h (Dos ri x rd) = f (foldI f g h ri) x (foldI f g h rd)
```

Define la función anterior a partir de este plegado.

```
enOrden = foldI_______________________________________________________
```