

A Prolog eljárás megírását kérő feladatokban a jegyzetben szereplő összes Prolog eljárás (akár beépített, akár a jegyzetben definiált) szabadon használható. Segédeljárást nem definiálhat.

Az egyes részfeladatokra azok szám- és betűjelével (pl. 2.b) hivatkozzon!

1. Döntse el, mi lesz az alábbi Prolog kérdések eredménye (hiba, meghiúsulás, siker)! Siker esetén adja meg a keletkező változó-behelyettesítéseket! A kérdéseket egyenként és önmagukban adjuk át az értelmezőknek. (5 pont)

(a)  $1+4 = X, \setminus + X = 5.$

(b)  $X \text{ is } Y*3, Y = 2+1.$

(c)  $[X|Y] = [1,2].$

(d)  $X \text{ is } 2+2, 2+2 = X.$

(e)  $X/Y = 24/12/3.$

2. Írja fel az alábbi kifejezések alapstruktúra-alakját (azaz szintaktikus édesítőszerek nélküli formáját), vagy rajzolja fel a hozzájuk tartozó fastruktúrákat! (6 pont)

(a)  $2-3+a*b$

(b)  $[3/4, [], X]$

(c)  $[1,2|[a-b/3|L]]$

3. Milyen változó-behelyettesítéseket eredményeznek az alábbi egyesítések? (9 pont)

(a)  $g(1+2+3, [a,b]) = g(X+Y, [U|V]).$

(b)  $[S, T|[T]] = .([1], [A*2, B*A]).$

(c)  $f(G*H, [2*3+b|J]) = f(C, [C+D, D]).$

4. Tekintse a p/1 eljárást és az azt hívó alábbi célsorozatot!

p(B) :-

```
    write(!),
    member(A, B),
    A =< 3,
    write(A),
    fail.
```

p(\_).

```
| ?- write(ÉÉHHNN), nl, p([É,É,H,H,N,N]), nl.
```

Mit ír ki a fenti célsorozat, ha abban az ÉÉHHNN karakterek helyén az ön születési dátumának karakterei (pl. 811231 ill. [8,1,1,2,3,1]) szerepel? Fogalmazza meg általánosan, hogy a p/1 által kiírt szöveg hogyan függ a paramétertől! (Futtatáskor a lists könyvtár be van töltve.) (5 pont)

5. Írjon olyan Prolog eljárást, amely megfelel az alábbi fejkomentnek! (5 pont)

```
% parospoz(+L, -A): A egy pozitív szám, amely páros indexű eleme az L
% számlistának. A listaelemeket 1-től számozzuk.
```

```
| ?- parospoz([1,2,4,-2,6,8], P).
      P = 2 ? ; P = 8 ? ; no
```

Az SML-függvény megírását kérő feladatokban a jegyzetben szereplő összes SML-függvény (akár beépített, akár a jegyzetben definiált) szabadon használható. Segédeljársást nem definiálhat. Az egyes részfeladatokra azok szám- és betűjelével (pl. 6.b) hivatkozzon!

A feladatokban előforduló könyvtári függvények típusa (az aritmetikai függvények kivételével):

List.filter	: ('a -> bool) -> 'a list -> 'a list	Char.isDigit	: char -> bool
foldl	: ('a * 'b -> 'b) -> 'b -> 'a list -> 'b	Char.isUpper	: char -> bool
foldr	: ('a * 'b -> 'b) -> 'b -> 'a list -> 'b	Char.toLower	: char -> char
map	: ('a -> 'b) -> 'a list -> 'b list	chr	: int -> char
op::	: 'a * 'a list -> 'a list	explode	: string -> char list
op^	: string * string -> string	implode	: char list -> string
rev	: 'a list -> 'a list	ord	: char -> int

6. Mi az **f típusa** az alábbi (egymástól független) deklarációkban? (8 pont)

- (a) fun f x y z = z(x y)
- (b) fun f (y, x) = y(ord x, 2.0)
- (c) fun f y x = y(x y)

7. Mi az **x értéke** és **típusa** az alábbi (egymástól független) deklarációkban? (3 pont)

- (a) val x = explode "maj" @ ["#o", "#m"]
- (b) val x = List.foldr op+ 10.0 [1.0, ~2.0, 3.0]
- (c) val x = map (fn i => if i < 3 then ~i else i) [2,4,3,1]

8. Mit kap eredményül, ha az alábbi **f** függvényt a hat számjegyből álló, **ÉÉHHNN** alakban füzéreként megadott **SAJÁT** születési dátumára alkalmazza (pl. f "720431")? Válaszát indokolja! (6 pont)

```
fun f x = let fun g (x::xs) (y::ys) = x :: y :: g ys xs | g _ _ = []
           val zs = explode x
           in implode(g zs (rev zs)) end
```

9. Írja meg az alábbi **f** függvény egy ekvivalens változatát a megadott könyvtári függvényekkel! (6 pont)

```
local fun f0 [] zs = rev zs
      | f0 (y::ys) zs =
          f0 ys (if #"0" <= y andalso y <= #"9" then ord y::zs else zs)
in fun f xs = f0 xs [] end
```

Az alkalmazandó függvények: List.filter, map, Char.isDigit, ord. Használja fel mindet! Segéd-függvényt NEM definiálhat!

10. Adott egy előjeles egész számokat tartalmazó lista, továbbá az alábbi adattípus-deklaráció. Írjon olyan függvényt **negPos** néven, amely egy 'a que típusú sort állít elő úgy, hogy a lista negatív elemei a sor elejére, nemnegatív elemei pedig a sor végére kerüljenek (tetszőleges sorrendben). A sor eleje, ill. vége: a <<, ill. >> adatkonstruktorral az adatszerkezetbe illesztett értékek sorozata. (7 pont)

```
infixr 5 <<; infix 5 >>; datatype 'a que = Nil | << of 'a * 'a que | >> of 'a que * 'a
(* negPos : int list -> 'a que
   negPos xs = olyan sor, amelynek az elején az xs negatív, a végén pedig az xs
   pozitív elemei vannak (tetszőleges sorrendben) *)
```

```
Példák: negPos [] = Nil
        negPos [~2, ~3, ~1] = ~2 << ~3 << ~1 << Nil
        negPos [2, 0, 1] = Nil >> 1 >> 0 >> 2
        negPos [~2, 2, ~1, 0, ~3, 1] = ~2 << ~1 << ~3 << (Nil >> 1 >> 0 >> 2)
```

A Prolog eljárás megírását kérő feladatokban a jegyzetben szereplő összes Prolog eljárás (akár beépített, akár a jegyzetben definiált) szabadon használható. Segédeljárást nem definiálhat.

Az egyes részfeladatokra azok szám- és betűjelével (pl. 2.b) hivatkozzon!

1. Döntse el, mi lesz az alábbi Prolog kérdések eredménye (hiba, meghiúsulás, siker)! Siker esetén adja meg a keletkező változó-behelyettesítéseket! A kérdéseket egyenként és önmagukban adjuk át az értelmezőknek. (5 pont)

- (a)  $[X| [Y]] = [a,b]$ .
- (b)  $X \text{ is } 2*3, \setminus + X = 6$ .
- (c)  $Y \text{ is } X+1, X = 1$ .
- (d)  $A-B = 10-5-2$ .
- (e)  $X = 2*3, Y = X+1$ .

2. Írja fel az alábbi kifejezések alapstruktúra-alakját (azaz szintaktikus édesítőszerek nélküli formáját), vagy rajzolja fel a hozzájuk tartozó fastruktúrákat! (6 pont)

- (a)  $u-v+1-2$
- (b)  $[1,a+b*c|X]$
- (c)  $[a,b-c+d/e, []]$

3. Milyen változó-behelyettesítéseket eredményeznek az alábbi egyesítések? (9 pont)

- (a)  $[X*Y, V|W] = [a*b*c, 4+5]$ .
- (b)  $[E*2*3, F*G, E] = .(A, [A, 1])$ .
- (c)  $g(H-G, [H, G]) = g(U*T-a, [T*2|S])$ .

4. Tekintse a p/1 eljárást és az azt hívó alábbi célsorozatot!

```
p(B) :-  
    member(A, B),  
    write(+),  
    A >= 3,  
    write(A),  
    fail.
```

```
p(_).
```

```
| ?- write(ÉÉHHNN), nl, p([É,É,H,H,N,N]), nl.
```

Mit ír ki a fenti célsorozat, ha abban az ÉÉHHNN karakterek helyén az ön születési dátumának karakterei (pl. 811231 ill. [8,1,1,2,3,1]) szerepel? Fogalmazza meg általánosan, hogy a p/1 által kiírt szöveg hogyan függ a paramétertől! (Futtatáskor a lists könyvtár be van töltve.) (5 pont)

5. Írjon olyan Prolog eljárást, amely megfelel az alábbi fejkommentnek! (5 pont)

```
% parosneg(+L, -A): A egy negatív szám, amely páros indexű eleme az L  
% számlistának. A listaelemeket 1-től számozzuk.
```

```
| ?- parosneg([1,-2,-4,0,-8,-10], P).  
      P = -2 ? ; P = -10 ? ; no
```

Az SML-függvény megírását kérő feladatokban a jegyzetben szereplő összes SML-függvény (akár beépített, akár a jegyzetben definiált) szabadon használható. Segédeljársást nem definiálhat. Az egyes részfeladatokra azok szám- és betűjelével (pl. 6.b) hivatkozzon!

A feladatokban előforduló könyvtári függvények típusa (az aritmetikai függvények kivételével):

List.filter	: ('a -> bool) -> 'a list -> 'a list	Char.isDigit	: char -> bool
foldl	: ('a * 'b -> 'b) -> 'b -> 'a list -> 'b	Char.isUpper	: char -> bool
foldr	: ('a * 'b -> 'b) -> 'b -> 'a list -> 'b	Char.toLower	: char -> char
map	: ('a -> 'b) -> 'a list -> 'b list	chr	: int -> char
op::	: 'a * 'a list -> 'a list	explode	: string -> char list
op^	: string * string -> string	implode	: char list -> string
rev	: 'a list -> 'a list	ord	: char -> int

6. Mi az **f típusa** az alábbi (egymástól független) deklarációkban? (8 pont)

- (a) fun f x y z = y(z x)
- (b) fun f (y, x) = x(1, chr y)
- (c) fun f x y = (y, (x, y))

7. Mi az **x értéke** és **típusa** az alábbi (egymástól független) deklarációkban? (3 pont)

- (a) val x = "zeb" ^ implode ["r", "a"]
- (b) val x = List.foldl op:: ["K", "U", "K"] ["A", "C"]
- (c) val x = List.filter (fn i => i >= 3) [2,4,3,1]

8. Mit kap eredményül, ha az alábbi **f** függvényt a hat számjegyből álló, ÉÉHHNN alakban füzéreként megadott SAJÁT születési dátumára alkalmazza (pl. f "720431")? Válaszát indokolja! (6 pont)

```
fun f x = let fun g (x::xs) (y::ys) = y :: x :: g ys xs | g _ _ = []
          val zs = explode x
          in implode(g zs (rev zs)) end
```

9. Írja meg az alábbi **f** függvény egy ekvivalens változatát a megadott könyvtári függvényekkel! (6 pont)

```
fun f xs =
  let fun f0 [] zs = rev zs
      | f0 (y::ys) zs =
          f0 ys (if #"A" <= y andalso y <= #"Z" then Char.toLower y::zs else zs)
      in f0 xs [] end
```

Az alkalmazandó függvények: List.filter, map, Char.toLower, Char.isUpper. Használja fel mindet! Segédfüggvényt NEM definiálhat!

10. Adott egy előjeles egész számokat tartalmazó lista, továbbá az alábbi adattípus-deklaráció. Írjon olyan függvényt oddEven néven, amely egy 'a que típusú sort állít elő úgy, hogy a lista páratlan elemei a sor elejére, páros elemei pedig a sor végére kerüljenek (tetszőleges sorrendben). A sor eleje, ill. vége: a <<, ill. >> adatkonstruktorral az adatszerkezetbe illesztett értékek sorozata. Egy szám páros/páratlan voltának meghatározására használhatja a fun even n = n mod 2 = 0 függvényt. (7 pont)

```
infixr 5 <<; infix 5 >>; datatype 'a que = Nil | << of 'a * 'a que | >> of 'a que * 'a
```

```
(* oddEven : int list -> 'a que
   oddEven xs = olyan sor, amelynek az elején az xs páratlan, a végén pedig az xs
   páros elemei vannak (tetszőleges sorrendben) *)
```

```
Példák: oddEven [] = Nil
         oddEven [5, ~3, 1] = 5 << ~3 << 1 << Nil
         oddEven [2, 0, ~4] = Nil >> ~4 >> 0 >> 2
         oddEven [5, 1, 2, ~3, ~4, 0] = 5 << 1 << ~3 << (Nil >> 0 >> ~4 >> 2)
```

A Prolog eljárás megírását kérő feladatokban a jegyzetben szereplő összes Prolog eljárás (akár beépített, akár a jegyzetben definiált) szabadon használható. Segédeljárást nem definiálhat.

Az egyes részfeladatokra azok szám- és betűjelével (pl. 2.b) hivatkozzon!

1. Döntse el, mi lesz az alábbi Prolog kérdések eredménye (hiba, meghiúsulás, siker)! Siker esetén adja meg a keletkező változó-behelyettesítéseket! A kérdéseket egyenként és önmagukban adjuk át az értelmezőknek. (5 pont)

(a)  $2+7 = A, \setminus + A = 9.$

(b)  $[U|V] = [a,b].$

(c)  $A \text{ is } 3+1, A = 3+1.$

(d)  $X*Y = 2*3+1.$

(e)  $U \text{ is } 2*2, V = U+3.$

2. Írja fel az alábbi kifejezések alapstruktúra-alakját (azaz szintaktikus édesítőszerek nélküli formáját), vagy rajzolja fel a hozzájuk tartozó fastruktúrákat! (6 pont)

(a)  $2/3+a+b$

(b)  $[3/4, a, X]$

(c)  $[u, v | [u-v*3 | X]]$

3. Milyen változó-behelyettesítéseket eredményeznek az alábbi egyesítések? (9 pont)

(a)  $h(A*B, [C|D]) = h(1*2*3, [u, v]).$

(b)  $.( [a], [E+2, F+E] ) = [Y, X | [X]].$

(c)  $g(G/H, [2/3+u|J]) = g(U, [U+V, V]).$

4. Tekintse a p/1 eljárást és az azt hívó alábbi célsorozatot!

p(B) :-

```
member(A, B),  
A >= 2,  
write(-),  
write(A),  
fail.
```

p(\_).

```
| ?- write(ÉÉHHNN), nl, p([É,É,H,H,N,N]), nl.
```

Mit ír ki a fenti célsorozat, ha abban az **ÉÉHHNN** karakterek helyén az ön születési dátumának karakterei (pl. 811231 ill. [8,1,1,2,3,1]) szerepel? Fogalmazza meg általánosan, hogy a p/1 által kiírt szöveg hogyan függ a paramétertől! (Futtatáskor a lists könyvtár be van töltve.) (5 pont)

5. Írjon olyan Prolog eljárást, amely megfelel az alábbi fejkomentnek! (5 pont)

```
% ptlanpoz(+L, -A): A egy pozitív szám, amely páratlan indexű eleme az L  
% számlistának. A listaelemeket 1-től számozzuk.
```

```
| ?- ptlanpoz([2,4,-2,6,8], P).
```

```
P = 2 ? ; P = 8 ? ; no
```

Az SML-függvény megírását kérő feladatokban a jegyzetben szereplő összes SML-függvény (akár beépített, akár a jegyzetben definiált) szabadon használható. Segédeljársást nem definiálhat. Az egyes részfeladatokra azok szám- és betűjelével (pl. 6.b) hivatkozzon!

A feladatokban előforduló könyvtári függvények típusa (az aritmetikai függvények kivételével):

List.filter	: ('a -> bool) -> 'a list -> 'a list	Char.isDigit	: char -> bool
foldl	: ('a * 'b -> 'b) -> 'b -> 'a list -> 'b	Char.isUpper	: char -> bool
foldr	: ('a * 'b -> 'b) -> 'b -> 'a list -> 'b	Char.toLower	: char -> char
map	: ('a -> 'b) -> 'a list -> 'b list	chr	: int -> char
op::	: 'a * 'a list -> 'a list	explode	: string -> char list
op^	: string * string -> string	implode	: char list -> string
rev	: 'a list -> 'a list	ord	: char -> int

6. Mi az **f típusa** az alábbi (egymástól független) deklarációkban? (8 pont)

- (a) fun f x y z = x(y z)
- (b) fun f (y, x) = y(chr x, #"B")
- (c) fun f x y = (y, (x y))

7. Mi az **x értéke** és **típusa** az alábbi (egymástól független) deklarációkban? (3 pont)

- (a) val x = (ord #"1" - ord #"0") :: [0, 1, 2+3]
- (b) val x = List.foldr op:: ["K", #"U", #"K"] ["A", #"C"]
- (c) val x = map (fn i => if i >= 3 then (true, i) else (false, i)) [2,4,3,1]

8. Mit kap eredményül, ha az alábbi **f** függvényt a hat számjegyből álló, ÉÉHHNN alakban füzéreként megadott SAJÁT születési dátumára alkalmazza (pl. f "720431")? Válaszát indokolja! (6 pont)

```
fun f x = let fun g (x::xs) (y::ys) = y :: x :: g (rev xs) ys | g _ _ = []
          val zs = explode x
          in implode(g zs (rev zs)) end
```

9. Írja meg az alábbi **f** függvény egy ekvivalens változatát a megadott könyvtári függvényekkel! (6 pont)

```
local fun f0 [] zs = rev zs
      | f0 (y::ys) zs =
          f0 ys (if #"0" <= chr y andalso chr y <= #"9" then chr y::zs else zs)
in fun f xs = f0 xs [] end
```

Az alkalmazandó függvények: List.filter, map, Char.isDigit, chr. Használja fel mindet! Segéd-függvényt NEM definiálhat!

10. Adott egy ascii kódolású kis- és nagybetűket tartalmazó lista, továbbá az alábbi adattípus-deklaráció. Írjon olyan függvényt lowUp néven, amely egy 'a que típusú sort állít elő úgy, hogy a listaelemek közül a kisbetűk a sor elejére, a nagybetűk pedig a sor végére kerüljenek (tetszőleges sorrendben). A sor eleje, ill. vége: a <<, ill. >> adatkonstruktorral az adatszerkezetbe illesztett értékek sorozata. Egy karakter kisbetű/nagybetű voltának meghatározására használhatja a Char.isUpper függvényt. (7 pont)

```
infixr 5 <<; infix 5 >>; datatype 'a que = Nil | << of 'a * 'a que | >> of 'a que * 'a
(* lowUp : char list -> 'a que
   lowUp xs = olyan sor, amelynek az elején az xs-beli kisbetűk, a végén pedig az
   xs-beli nagybetűk vannak (tetszőleges sorrendben) *)
```

```
Példák: lowUp [] = Nil
lowUp ["c", #"z", #"e"] = #"c" << #"z" << #"e" << Nil
lowUp ["C", #"Z", #"E"] = Nil >> #"E" >> #"Z" >> #"C"
lowUp ["c", #"C", #"E", #"z", #"e", #"Z"] =
    #"c" << #"z" << #"e" << (Nil >> #"Z" >> #"E" >> #"C")
```

A Prolog eljárás megírását kérő feladatokban a jegyzetben szereplő összes Prolog eljárás (akár beépített, akár a jegyzetben definiált) szabadon használható. Segédeljárást nem definiálhat.

Az egyes részfeladatokra azok szám- és betűjelével (pl. 2.b) hivatkozzon!

1. Döntse el, mi lesz az alábbi Prolog kérdések eredménye (hiba, meghiúsulás, siker)! Siker esetén adja meg a keletkező változó-behelyettesítéseket! A kérdéseket egyenként és önmagukban adjuk át az értelmezőknek. (5 pont)

- (a)  $X*Y = 2*3*4$ .
- (b)  $X \text{ is } 8+2, \setminus + X = 10$ .
- (c)  $X = Y+2, Y \text{ is } 2*2$ .
- (d)  $[P|[Q]] = [a,b]$ .
- (e)  $9+5+2 \text{ is } A+B$ .

2. Írja fel az alábbi kifejezések alapstruktúra-alakját (azaz szintaktikus édesítőszerek nélküli formáját), vagy rajzolja fel a hozzájuk tartozó fastruktúrákat! (6 pont)

- (a)  $u/v+1*2$
- (b)  $[1+2+3,4,X]$
- (c)  $[u,i*j|[[]],d/e]$

3. Milyen változó-behelyettesítéseket eredményeznek az alábbi egyesítések? (9 pont)

- (a)  $[p+q+r, 5+8] = [X+Y, Z|V]$ .
- (b)  $.(U, [U, 1]) = [E+2+3, F+G, E]$ .
- (c)  $f(W/P-a, [P/4|Q]) = f(H-G, [H, G])$ .

4. Tekintse a p/1 eljárást és az azt hívó alábbi célsorozatot!

```
p(B) :-  
    member(A, B),  
    A =< 4,  
    write(A),  
    write(*),  
    fail.
```

```
p(_).
```

```
| ?- write(ÉÉHHNN), nl, p([É,É,H,H,N,N]), nl.
```

Mit ír ki a fenti célsorozat, ha abban az *ÉÉHHNN* karakterek helyén az ön születési dátumának karakterei (pl. 811231 ill. [8,1,1,2,3,1]) szerepel? Fogalmazza meg általánosan, hogy a p/1 által kiírt szöveg hogyan függ a paramétertől! (Futtatáskor a *lists* könyvtár be van töltve.) (5 pont)

5. Írjon olyan Prolog eljárást, amely megfelel az alábbi fejkomentnek! (5 pont)

```
% ptlanneg(+L, -A): A egy negatív szám, amely páratlan indexű eleme az L  
% számlistának. A listaelemeket 1-től számozzuk.
```

```
| ?- ptlanneg([-2,-4,0,-8,-10], P).  
      P = -2 ? ; P = -10 ? ; no
```

Az SML-függvény megírását kérő feladatokban a jegyzetben szereplő összes SML-függvény (akár beépített, akár a jegyzetben definiált) szabadon használható. Segédeljársást nem definiálhat. Az egyes részfeladatokra azok szám- és betűjelével (pl. 6.b) hivatkozzon!

A feladatokban előforduló könyvtári függvények típusa (az aritmetikai függvények kivételével):

List.filter	: ('a -> bool) -> 'a list -> 'a list	Char.isDigit	: char -> bool
foldl	: ('a * 'b -> 'b) -> 'b -> 'a list -> 'b	Char.isUpper	: char -> bool
foldr	: ('a * 'b -> 'b) -> 'b -> 'a list -> 'b	Char.toLower	: char -> char
map	: ('a -> 'b) -> 'a list -> 'b list	chr	: int -> char
op::	: 'a * 'a list -> 'a list	explode	: string -> char list
op^	: string * string -> string	implode	: char list -> string
rev	: 'a list -> 'a list	ord	: char -> int

6. Mi az **f típusa** az alábbi (egymástól független) deklarációkban? (8 pont)

- (a) fun f x y z = z(y x)
- (b) fun f (y, x) = y(rev x, 1)
- (c) fun f x y = (y, (y, x))

7. Mi az **x értéke** és **típusa** az alábbi (egymástól független) deklarációkban? (3 pont)

- (a) val x = [0, 1, 2+3] @ [ord #"5" - ord #"0"]
- (b) val x = List.foldl op^ ("K" ^ "U" ^ "K") ["A", "C"]
- (c) val x = List.filter (fn c => ord c mod 2 = 0) [chr 48, chr 49, chr 50, chr 51]

8. Mit kap eredményül, ha az alábbi **f** függvényt a hat számjegyből álló, ÉÉHHNN alakban füzéreként megadott SAJÁT születési dátumára alkalmazza (pl. f "720431")? Válaszát indokolja! (6 pont)

```
fun f x = let fun g (x::xs) (y::ys) = y :: x :: g (rev ys) xs | g _ _ = []
          val zs = explode x
          in implode(g zs zs) end
```

9. Írja meg az alábbi **f** függvény egy ekvivalens változatát a megadott könyvtári függvényekkel! (6 pont)

```
fun f xs = let fun f0 [] zs = rev zs
              | f0 (y::ys) zs =
                f0 ys (if #"A" <= chr y andalso chr y <= #"Z" then chr y::zs else zs)
              in f0 xs [] end
```

Az alkalmazandó függvények: List.filter, map, Char.isUpper, chr. Használja fel mindet! Segéd-függvényt NEM definiálhat!

10. Adott egy ascii kódolású karaktereket tartalmazó lista, továbbá az alábbi adattípus-deklaráció. Írjon olyan függvényt digNondig néven, amely egy 'a que típusú sort állít elő úgy, hogy a listaelemek közül a számjegyek a sor elejére, az egyéb karakterek pedig a sor végére kerüljenek (tetszőleges sorrendben). A sor eleje, ill. vége: a <<, ill. >> adatkonstruktorral az adatszerkezetbe illesztett értékek sorozata. Egy karakter számjegy voltának meghatározására használhatja a Char.isDigit függvényt. (7 pont)

```
infixr 5 <<; infix 5 >>; datatype 'a que = Nil | << of 'a * 'a que | >> of 'a que * 'a
```

```
(* digNondig : char list -> 'a que
   digNondig xs = olyan sor, amelynek az elején az xs-beli számjegyek, a végén pedig
   az xs többi karakterei vannak (tetszőleges sorrendben) *)
```

Példák: digNondig [] = Nil  
 digNondig ["9", "5", "4"] = "9" << "5" << "4" << Nil  
 digNondig ["C", "?", ":"] = Nil >> ":" >> "?" >> "C"  
 digNondig ["9", "C", ":", "4", "5", "?"] =  
 "9" << "4" << "5" << (Nil >> "?" >> ":" >> "C")