Sortarea listelor de 1, 2, 3:

Algebra specificată este: $\mathcal{A} = \langle E, L, \bar{\ }, \cdot, 1, 2, 3 \rangle$ cu:

$E = \{1, 2, 3\}$
$L = \{s \in E^+ | \forall i \in \overline{1, |s| - 1}\, s_i \leq s_{i+1}\}$
$\bar{\ } : E \to L$, $\bar{i}$ =lista formată din i
$\cdot : L \times L \to L$, $s \cdot s'$ =interclasarea lui s cu s'.

Obs. că $\cdot$ este şi asociativă şi comutativă.

Specificaţie:

Signatura:

Sorturi: e, l

Simboluri de operaţii: $\{\} : e \to l,\ \ , : l\, l \to l,\ \ 1\, 2\, 3 : \to l.$

Ecuaţii:

$\{2\}, \{1\} = \{1\}, \{2\}$
$\{3\}, \{1\} = \{1\}, \{3\}$
$\{3\}, \{2\} = \{2\}, \{3\}$
$(\forall X, Y, Z : l)\ (X, Y), Z = X, (Y, Z)$.

Demonstrăm că $\mathcal{A}$ este model iniţial pentru specificaţia dată:

- Este clar că $\mathcal{A}$ satisface specificaţia (reamintim că $\cdot$ este asociativă).
- Demonstrăm că dacă $\mathcal{B}$ este un alt model pentru specificaţie atunci $\exists!$ morfism de la $\mathcal{A}$ la $\mathcal{B}$.

  Fie $\mathcal{B} = \langle B_e, B_l, \bar{\ }^B, ,_B, 1_B, 2_B, 3_B \rangle$ un model.

  !: Fie $f, g : \mathcal{A} \to \mathcal{B}$ morfisme.

  - Pentru orice $i \in E$ avem $f_e(i) = i_B = g_e(i)$ (deoarece f, g morfisme)
  - Demonstrăm că pentru orice $s \in L$ avem $f_l(s) = g_l(s)$, prin inducţie după lungimea lui s:
    - dacă $s = \bar{i}$, $i \in E$, atunci $f_l(s) = f_l(\bar{i}) \stackrel{f\,mf}{=} \overline{f_e(i)}^B = \overline{i_B}^B = \ldots = g_l(s)$;
    - dacă $s = is'$, $i \in E$ şi $f_l(s') = g_l(s')$, atunci $f_l(s) = f_l(is) = f_l(\bar{i} \cdot s') \stackrel{f\,mf}{=} f_l(\bar{i}) ,_B f_l(s') \stackrel{ip\,ind}{=} \overline{i_B}^B ,_B g_l(s') = \ldots = g_l(s)$.

  $\exists$: Fie $f : \mathcal{A} \to B$ definită prin:

  $f_e(i) = i_B$, $\forall i \in E$
  $f_l(\bar{i}) = \overline{i_B}^B$, $\forall i \in E$
  $f_l(is) = \overline{i_B}^B ,_B f_l(s)$, $\forall i \in E$ şi $\forall s \in L$ a.î. $f_l(s)$ e definit

  (def. e corectă, deoarece L e generat de $\{\bar{i} | i \in E\}$ şi orice $s \in L$ cu $|s| > 1$ se scrie unic sub forma $s = is'$ cu $i \in E$, $s' \in L$ şi $|s'| < |s|$).

  Demonstrăm că f este morfism:

  - pentru orice $i \in E$ avem $f_e(i) = i_B$, conform definiţiei;
  - pentru orice $i \in E$ avem $f_l(\bar{i}) \stackrel{cf\,def}{=} \overline{i_B}^B = \overline{f_e(i)}^B$;
  - demonstrăm că pentru orice $s, s' \in L$ avem $f_l(s \cdot s') = f_l(s) ,_B f_l(s')$, prin inducţie:

- dacă $i, j \in E$ şi $i \leq j$, atunci:
  $f_l(\bar{i} \cdot \bar{j}) = f_l(ij) \overset{cf\,def}{=} \overline{i_B}^B ,_B f_l(\bar{j}) = \overline{i_B}^B ,_B \overline{j_B}^B = f_l(\bar{i}) ,_B f_l(\bar{j})$;
- dacă $i, j \in E$ şi $i > j$, atunci:
  $f_l(\bar{i} \cdot \bar{j}) = f_l(ji) = \overline{j_B}^B ,_B \overline{i_B}^B \overset{ec\,din\,spec}{=} \overline{i_B}^B ,_B \overline{j_B}^B = f_l(\bar{i}) ,_B f_l(\bar{j})$;
- dacă $i, j \in E$ şi $s \in L$ a.î. $i \leq j$ şi $f_l(\bar{j} \cdot s) = f_l(\bar{j}) ,_B f_l(s)$ atunci:
  $f_l(\bar{i} \cdot js) = f_l(ijs) \overset{cf\,def}{=} \overline{i_B}^B ,_B f_l(js) = f_l(\bar{i}) ,_B f_l(js)$;
- dacă $i, j \in E$ şi $s \in L$ a.î. $i > j$ şi $f_l(\bar{j} \cdot s) = f_l(\bar{j}) ,_B f_l(s)$ atunci:
  $f_l(\bar{i} \cdot js) = f_l(j(\bar{i} \cdot s))$ (argumentul este şirul care începe cu j şi continuă cu interclasarea dintre $\bar{i}$ şi s) $\overset{cf\,def}{=} \overline{j_B}^B ,_B f_l(\bar{i} \cdot s) \overset{ip\,ind}{=} \overline{j_B}^B ,_B (f_l(\bar{i}) ,_B f_l(s)) = \overline{j_B}^B ,_B (\overline{i_B}^B ,_B f_l(s)) \overset{,\,asoc}{=} (\overline{j_B}^B ,_B \overline{i_B}^B) ,_B f_l(s) \overset{ip\,ind}{=} f_l(\bar{j} \cdot \bar{i}) ,_B f_l(s) \overset{\cdot\,com}{=} f_l(\bar{i} \cdot \bar{j}) ,_B f_l(s) \overset{ip\,ind}{=} (f_l(\bar{i}) ,_B f_l(\bar{j})) ,_B f_l(s) \overset{,\,asoc}{=} f_l(\bar{i}) ,_B (f_l(\bar{j}) ,_B f_l(s)) = f_l(\bar{i}) ,_B (\overline{j_B}^B ,_B f_l(s)) = f_l(\bar{i}) ,_B f_l(js)$;
- dacă $i \in E$, $s, s' \in L$ a.î. $f_l(s \cdot s') = f_l(s) ,_B f_l(s')$, atunci:
  $f_l(is \cdot s') = f_l((\bar{i} \cdot s) \cdot s') \overset{\cdot\,asoc}{=} f_l(\bar{i} \cdot (s \cdot s')) \overset{ip\,ind}{=} f_l(\bar{i})'_B f_l(s \cdot s') = \overline{i_B}^B ,_B (f_l(s) ,_B f_l(s')) \overset{,\,asoc}{=} (\overline{i_B}^B ,_B f_l(s)) ,_B f_l(s')) = f_l(is) ,_B f(s')$.

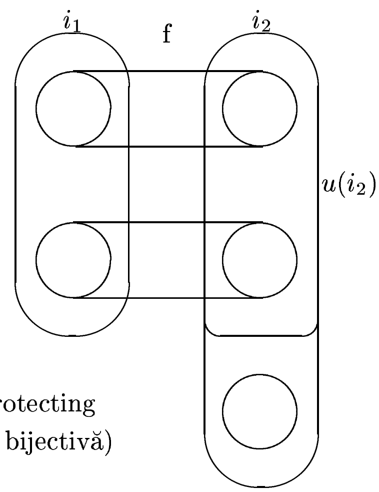
Deci specificaţia caracterizeaza pe $\mathcal{A}$ ca tip abstract de date.

Sistemul de rescriere definit de ea este clar terminatoriu (orice rescriere se termină după un nr. finit de paşi într-o listă sortată) şi confluent (orice termen se rescrie, indiferent de cale, in lista cu aceleaşi elemente dar sortată), deci programul OBJ se poate obtine transcriind direct specificaţia.

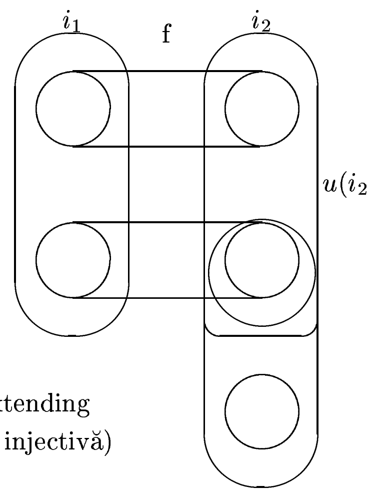
Observaţii:

1. Ceea ce în specificaţie şi programul OBJ părea a fi simpla operaţie de concatenare, modelează de fapt interclasarea din algebra de liste sortate.
2. Va rezulta că operatorul , din specificaţie satisface şi comutativitatea (pentru că operaţia modelată $\cdot$ o satisface).

$$
\begin{array}{ccc}
 & \mathrm{M}_1 \quad \subseteq \quad \mathrm{M}_2 & \\
(\Sigma_1, E_1) & \xhookrightarrow{\qquad\qquad i \qquad\qquad} & (\Sigma_2, E_2) \\
i_1\ (\text{iniţial}),\ u(i_2) \ \in \ \mathcal{A}lg_{\Sigma_1,E_1} & \xleftarrow{\qquad\qquad u \qquad\qquad} & \mathcal{A}lg_{\Sigma_2,E_2} \ \ni \ i_2\ (\text{iniţial}) \\
u(i_2) & \xleftarrow{\qquad\qquad\qquad\qquad} & i_2 \\
 & & \\
i_1 & \xrightarrow{\qquad\qquad \exists! f \qquad\qquad} & u(i_2)
\end{array}
$$



protecting
(f bijectivă)



extending
(f injectivă)

```
obj M0 is           | obj M1 is           | obj M2 is                         | obj M3 is
sort Z .            | protecting M0 .     | extending M0 .                    | including M0 .
ops 0 1 : -> Z .    | sort B .            | ops c d : -> Z .                  | eq 1 = 0 .
op _+_ : Z Z -> Z . | ops t f : -> B .    | eq c + 1 = d . eq 1 + c = d .     | endo
var X : Z .         | op _<=_ : Z Z -> B .| eq d + 1 = c . eq 1 + d = c .     |
eq 0 + X = X .      | var X : Z .         | eq c + c = 1 . eq d + d = 1 .     |
eq X + 0 = X .      | eq 0 <= X = t .     | eq c + d = 0 . eq d + c = 0 .     |
eq 1 + 1 = 0 .      | eq X <= 1 = t .     | endo                              |
endo                | eq 1 <= 0 = f .     |                                   |
                    | endo                |                                   |
```

Tipul de date specificat de M0 este: $\mathcal{A} = \langle Z_2, 0, 1, +\rangle$ (exerciţiu).
Tipul de date specificat de M1 este: $\mathcal{B}_1 = \langle Z_2, \{t, f\}, 0, 1, t, f, +, <=\rangle$ (exerciţiu).
Tipul de date specificat de M2 este: $\mathcal{B}_2 = \langle\{\hat{0}, 0\hat{.}5, \hat{1}, 1\hat{.}5\}, \hat{0}, \hat{1}, 0\hat{.}5, 1\hat{.}5, +\rangle$ (exerciţiu).
Tipul de date specificat de M3 este: $\mathcal{B}_3 = \langle\{*\}, 0 := *, 1 := *, +\rangle$ (exerciţiu).
Avem următoarele ”uitări”:
$u_{10}(\mathcal{B}_1) = \langle Z_2, 0, 1, +\rangle = \mathcal{A}$
$u_{20}(\mathcal{B}_2) = \langle\{\hat{0}, 0\hat{.}5, \hat{1}, 1\hat{.}5\}, \hat{0}, \hat{1}, +\rangle$
$u_{30}(\mathcal{B}_3) = \langle\{*\}, 0 := *, 1 := *, +\rangle = \mathcal{B}_3$.
Următoarele morfisme $\exists!$:
$f_{01} : \mathcal{A} \to u_{10}(\mathcal{B}_1),\ f_{01}(0) = 0,\ f_{01}(1) = 1 \ \Rightarrow\ f$ bijectivă, incluziune protecting
$f_{02} : \mathcal{A} \to u_{20}(\mathcal{B}_2),\ f_{02}(0) = \hat{0},\ f_{02}(1) = \hat{1} \ \Rightarrow\ f$ injectivă, incluziune extending
$f_{03} : \mathcal{A} \to u_{30}(\mathcal{B}_3),\ f_{03}(0) = f_{03}(1) = * \ \Rightarrow\ f$ neinjectivă, incluziune including sau using.
Observaţii:

- Dacă adăugăm la M1:

  ```
  var A B C : Z .
  eq A + (B + C) = (A + B) + C .
  ```

  incluziunea ramâne protecting.

- Dacă adăugăm la M1:

  ```
  op _*_ : Z Z : -> Z .
  eq 0 * X = 0 . eq X * 0 = 0 . eq 1 * 1 = 1 .
  ```

  incluziunea ramâne protecting.

  Atunci obţinem $\mathcal{B}'_1 = \langle Z_2, \{t, f\}, 0, 1, t, f, +, *, <=\rangle$ şi $u_{10}(\mathcal{B}'_1) = u_{10}(\mathcal{B}_1)$.

- Daca adaugăm la M2 op. de scădere, incluziunea rămâne tot extending.

- Daca adaugăm la M2:
  eq c + d = d + c .
  incluziunea ramâne extending.

- Daca adaugăm la M2:
  eq 0 = c . eq 1 = c .
  in ideea de a transforma incluziunea în ”including”, de fapt pierdem confluenţa, deoarece:
  $0 + d \rightarrow d$
  $0 + d \rightarrow c + d \rightarrow 0 \rightarrow c$
  şi nu putem unifica c cu d. Deci procedeul de la M3 nu are mereu efectul dorit.

Sortarea listelor de numere naturale (varianta fără subsorturi), schiţa demonstraţiei:

Algebra specificată este: $\mathcal{A} = \langle I\!N, L, \{false, true\}, \bar{\ }, \cdot, alte\, op.(>)\rangle$ cu:

$L = \{s \in I\!N^+ | \forall i \in \overline{1, |s| - 1}\, s_i \leq s_{i+1}\}$

$\bar{\ } : I\!N \to L$, $\bar{i}$ =lista formată din i

$\cdot : L \times L \to L$, $s \cdot s'$ =interclasarea lui s cu s'.

Obs. că $\cdot$ este şi asociativă şi comutativă.

Specificaţie:

Signatura:

Sorturi: Nat, l, Bool

Simboluri de operaţii: $\{\} : e \to l$, $\ , : l\, l \to l$, alte op. importate.

Ecuaţii:

$(\forall X, Y : Nat)\, \{X\}, \{Y\} = \{Y\}, \{X\} \qquad if \qquad (X > Y)$

$(\forall X, Y, Z : l)\, (X, Y), Z = X, (Y, Z)$.

alte ec. importate

Demonstrăm că $\mathcal{A}$ este model iniţial pentru specificaţia dată:

- Este clar că $\mathcal{A}$ satisface specificaţia (reamintim că $\cdot$ este asociativă).
  Cum justificăm că satisface ecuaţia condiţională:
  Trebuie ca $\forall h : \{X, Y\} \to A_{Nat} = I\!N$ a.î. $h^\sharp_{Bool}(X > Y) = true_A = true$ să avem $h^\sharp_l(\{X\}, \{Y\}) = h^\sharp_l(\{Y\}, \{X\})$.

  Dar dacă $h^\sharp_{Bool}(X > Y) = true$, atunci $h(X) > h(Y)$ (pentru că $h^\sharp$ este morfism) deci $h^\sharp_l(\{X\}, \{Y\}) \overset{h^\sharp mf}{=} h^\sharp_l(\{X\}) \cdot h^\sharp_l(\{Y\}) \overset{h^\sharp mf}{=} \overline{h(X)} \cdot \overline{h(Y)} = \overline{h(Y)} \cdot \overline{h(X)}$ (deoarece $h(X) > h(Y)$ iar $\cdot$ = interclasare) $= \ldots = h^\sharp_l(\{Y\}, \{X\})$.

- Demonstrăm că dacă $\mathcal{B}$ este un alt model pentru specificaţie atunci $\exists!$ morfism de la $\mathcal{A}$ la $\mathcal{B}$.

  Fie $\mathcal{B} = \langle B_{Nat}, B_l, B_{Bool}, \bar{\ }^B, ,_B, etc.\rangle$ un model.

  !: Fie $f, g : \mathcal{A} \to \mathcal{B}$ morfisme. Atunci restrictţiile lor $f', g' : \mathcal{A}' := \langle I\!N, \{false, true\}, op.\, importate\rangle \to \mathcal{B}' := \langle B_{Nat}, B_{Bool}, op.\, coresp.\rangle$ sunt morfisme.

  Întrucât modulul NAT specifică $\mathcal{A}'$ ca tip abstract de date, rezultă că $f' = g'$, adică f şi g coincid pe $I\!N$şi $\{false, true\}$ şi mai mult, ştim şi forma lor:
  $\forall n \in I\!N\, f_{Nat}(n) = g_{Nat}(n) = s^n_B 0_B$, $f_{Bool}(false) = g_{Bool}(false) = false_B$, $f_{Bool}(true) = g_{Bool}(true) = true_B$

  Faptul că $\forall s \in L$ avem $f_l(s) = g_l(s)$ se dem. ca mai înainte prin inducţie după s, considerând cazurile:

  - $s = \bar{i}$, $i \in I\!N$
  - $s = is'$, $i \in I\!N$, $s \in L$ şi $f_l(s') = g_l(s')$.

  $\exists$: Fie $f : \mathcal{A} \to B$ definită prin:

  $f_{Nat}(n) := s^n_B 0_B$, $\forall n \in I\!N$;

  $f_{Bool}(false) := false_B$, $f_{Bool}(true) := true_B$;

  f_l se def. ca mai înainte inductiv:

$f_l(\bar{i}) = \overline{f_{Nat}(i)}^B = \overline{s_B^n 0_B}^B$, $\forall i \in I\!N$

$f_l(is) = \overline{s_B^n 0_B}^B ,_B f_l(s)$, $\forall i \in I\!N$ şi $\forall s \in L$ a.î. $f_l(s)$ e definit.

Faptul că f este morfism pt. operaţiile importate pe $I\!N$şi $\{false, true\}$ rezultă din proprietăţile modulelor NAT şi BOOL. Rămâne de arătat căf comută cu _ şi $\cdot$:

- pentru orice $i \in I\!N$ avem $f_l(\bar{i}) = \overline{f_{Nat}(i)}^B$, conform definiţiei;
- pentru orice $s, s' \in L$ avem $f_l(s \cdot s') = f_l(s),_B f_l(s')$ ca mai înainte prin inducţie, considerând cazurile:
  * $f_l(\bar{i} \cdot \bar{j})$, cu $i, j \in I\!N$, $i \leq j$;
  * $f_l(\bar{i} \cdot \bar{j})$, cu $i, j \in I\!N$, $i > j$;
  * $f_l(\bar{i} \cdot js)$, cu $i, j \in I\!N$, $s \in L$, $i \leq j$ şi $f_l(\bar{j} \cdot s) = f_l(\bar{j}),_B f_l(s)$;
  * $f_l(\bar{i} \cdot js)$, cu $i, j \in I\!N$, $s \in L$, $i > j$ şi $f_l(\bar{j} \cdot s) = f_l(\bar{j}),_B f_l(s)$;
  * $f_l(is, s')$, cu $i \in I\!N$, $s, s' \in L$ şi $f_l(s \cdot s') = f_l(s),_B f_l(s')$.

Detaliile: exercitiu (atenţie la demonstraţiile referitoare le ecuaţia condiţională).