Programozás
4. előadás

Programozási alapismeretek

- További programozási tételek
- Másolás – függvényszámítás
- Kiválogatás
- Szétválogatás
- Programozási tételek – visszatekintés
- Mátrixok

További programozási tételek

Mi az, hogy programozási tétel?

Típusfeladat általános megoldása.

- Sorozat → érték
- Sorozat → sorozat
- Sorozat → sorozatok
- Sorozatok → sorozat

7. Másolás – függvényszámítás

Feladatok:

- Egy számsorozat tagjainak adjuk meg az abszolút értékét!
- Egy szöveget alakítsunk át csupa kisbetűssé!
- Számoljuk ki két vektor összegét!
- Készítsünk függvénytáblázatot a sin(x) függvényről!
- Ismerünk N dátumot ’éé.hh.nn’ alakban, adjuk meg őket ’éé. hónapnév nn.’ alakban!

7. Másolás – függvényszámítás

Feladatok:

- Egy számsorozat tagjainak adjuk meg az abszolút értékét!
- Egy szöveget alakítsunk át csupa kisbetűssé!
- Számoljuk ki két vektor összegét!
- Készítsünk függvénytáblázatot a sin(x) függvényről!
- Ismerünk N dátumot ‚éé.hh.nn' alakban, adjuk meg ‚éé. hónapnév nn' alakban!

Mi bennük a közös?

N darab „valamihez" kell hozzárendelni másik N darab „valamit", ami akár az előbbitől különböző típusú is lehet. A darabszám, a sorrend is marad. Az elemeken operáló függvény ugyanaz.

7. Másolás – függvényszámítás

N darab „valamihez” kell hozzárendelni másik N darab „valamit”, ami akár az előbbitől különböző típusú is lehet. A darabszám marad, a sorrend is marad.

Specifikáció:

- Bemenet: $N \in \mathbb{N}$
  $X_{1..N} \in H_1^N$
  $f: H_1 \rightarrow H_2$
- Kimenet: $Y_{1..N} \in H_2^N$
- Előfeltétel: –
- Utófeltétel: $\forall i(1 \leq i \leq N): Y_i = f(X_i)$

  Másként: $Y_{1..N} = f(X_{1..N})$

7. Másolás – függvényszámítás

Algoritmus:

Specifikáció:

- Bemenet: $N \in \mathbb{N}$, $X_{1..N} \in H_1^N$, $f: H_1 \to H_2$
- Kimenet: $Y_{1..N} \in H_2^N$
- Előfeltétel: –
- Utófeltétel: $\forall i(1 \le i \le N)$: $Y_i = f(X_i)$

i=1..N

Y[i]:=f(X[i])

Változó
i:Egész

Megjegyzés: nem feltétlenül kell ugyanaz az i index a két tömbhöz, pl.:

Utófeltétel: $\forall i(1 \le i \le N)$: $Y_{p(i)} = f(X_i)$

i=1..N

Y[p(i)]:=f(X[i])

Változó
i:Egész

p(i) lehet pl. 2*i, N–i+1, … (megfelelő Y tömb mérettel, ill. indexintervallummal definiálva; Y részsorozata a kimenet; p injektív)

7. Másolás – függvényszámítás

N darab „valamihez” kell hozzárendelni másik N darab „valamit”, ami akár az előbbitől különböző típusú is lehet. A darabszám marad, a sorrend is marad.

Specifikáció (egy gyakori speciális eset)$_1$:

- Bemenet: $N \in \mathbb{N}$
  $X_{1..N} \in H^N$
  $g: H \to H$
  $T: H \to L$
- Kimenet: $Y_{1..N} \in H^N$
- Előfeltétel: –
- Utófeltétel: $\forall i(1 \leq i \leq N)$: $Y_i = f(X_i)$
- Definíció: $f(x) := \begin{cases} g(x), & \text{ha } T(x) \\ x, & \text{egyébként} \end{cases}$

$f: H \to H$

Specifikáció:

- Bemenet: $N \in \mathbb{N}$
  $X_{1..N} \in H_1^N$
  $f: H_1 \to H_2$
- Kimenet: $Y_{1..N} \in H_2^N$
- Előfeltétel: –
- Utófeltétel: $\forall i(1 \leq i \leq N)$: $Y_i = f(X_i)$

7. Másolás – függvényszámítás

Specifikáció (egy gyakori speciális eset)$_1$:

N darab „valamihez” kell hozzárendelni másik N darab „valamit”, ami akár az előbbitől különböző típusú is lehet. A darabszám marad, a sorrend is marad.

Specifikáció:

- Bemenet: $N \in \mathbb{N}$
  $X_{1..N} \in H_1^N$
  $f: H_1 \rightarrow H_2$
- Kimenet: $Y_{1..N} \in H_2^N$
- Előfeltétel: –
- Utófeltétel: $\forall i(1 \leq i \leq N)$: $Y_i = f(X_i)$

- Bemenet: $N \in \mathbb{N}$
  $X_{1..N} \in H^N$
  $g: H \rightarrow H$
  $T: H \rightarrow L$
- Kimenet: $Y_{1..N} \in H^N$
- Előfeltétel: –
- Utófeltétel: $\forall i(1 \leq i \leq N)$:

$$(T(X_i) \rightarrow Y_i = g(X_i) \quad \text{és}$$
$$\text{nem } T(X_i) \rightarrow Y_i = X_i)$$

7. Másolás – függvényszámítás

Algoritmus$_1$:

Specifikáció (egy gyakori speciális eset)$_1$:

- Bemenet: $N \in \mathbb{N}$
  $X_{1..N} \in H^N$
  $g: H \to H$
  $T: H \to L$
- Kimenet: $Y_{1..N} \in H^N$
- Előfeltétel: –
- Utófeltétel: $\forall i(1 \leq i \leq N)$:
  $(T(X_i) \to Y_i = g(X_i)$ és
  nem $T(X_i) \to Y_i = X_i)$

i=1..N

T(X[i])

I | N

Y[i]:=g(X[i]) | Y[i]:=X[i]

Változó
i:Egész

N darab „valamihez” kell hozzárendelni másik N darab „valamit”, ami akár az előbbitől különböző típusú is lehet. A darabszám marad, a sorrend is marad.

7. Másolás – függvényszámítás

Specifikáció (egy másik speciális eset)$_2$**:**

- Bemenet: $N \in \mathbb{N}$
  $X_{1..N} \in \mathbb{H}^N$
- Kimenet: $Y_{1..N} \in \mathbb{H}^N$
- Előfeltétel: –
- Utófeltétel: $\forall i(1 \leq i \leq N)$: $Y_i = X_i$

N darab „valamihez” kell hozzárendelni másik N darab „valamit”, ami akár az előbbitől különböző típusú is lehet. A darabszám marad, a sorrend is marad.

Specifikáció:

- Bemenet: $N \in N$
  $X_{1..N} \in H_1^N$
  $f: H_1 \rightarrow H_2$
- Kimenet: $Y_{1..N} \in H_2^N$
- Előfeltétel: –
- Utófeltétel: $\forall i(1 \leq i \leq N)$: $Y_i = f(X_i)$

Megjegyzés:

nincs f függvény, helyesebben identikus (f(x):=x).

7. Másolás – függvényszámítás

Algoritmus$_2$:

Specifikáció:

- Bemenet: $N \in \mathbb{N}$
  $X_{1..N} \in H_1^N$
  $f: H_1 \rightarrow H_2$
- Kimenet: $Y_{1..N} \in H_2^N$
- Előfeltétel: –
- Utófeltétel: $\forall i(1 \leq i \leq N): Y_i = f(X_i)$

i=1..N
Y[i]:=X[i]

Változó
i:**Egész**

Megjegyzés:

Az Y:=X értékadással helyettesíthető, ha a két tömb azonos típusú. Ha az indexek különbözőek (p nem identikus), akkor:

i=1..N
Y[p(i)]:=X[i]

Változó
i:**Egész**

7. Másolás – függvényszámítás

Specifikáció: $f(X[i]) \rightarrow P[i]+Q[i]$

- Bemenet: $N \in \mathbb{N}$
  $P_{1..N}, Q_{1..N} \in \mathbb{R}^N$
  $f: \mathbb{R} \times \mathbb{R} \rightarrow \mathbb{R},\ f(x,y) := x+y$
- Kimenet: $R_{1..N} \in \mathbb{R}^N$
- Előfeltétel: –
- Utófeltétel: $\forall i(1 \leq i \leq N): R_i = P_i + Q_i$

- Számoljuk ki két vektor összegét!

$(P,Q) \in (\mathbb{R} \times \mathbb{R})^N$

Specifikáció:

- Bemenet: $N \in \mathbb{N}$
  $X_{1..N} \in H_1^N$
  $f: H_1 \rightarrow H_2$
- Kimenet: $Y_{1..N} \in H_2^N$
- Előfeltétel: –
- Utófeltétel: $\forall i(1 \leq i \leq N): Y_i = f(X_i)$

Algoritmus:

Algoritmus:

i=1..N
Y[i]:=f(X[i])

i=1..N
R[i]:=P[i]+Q[i]

Változó
i:Egész

8. Kiválogatás

Feladatok:

- Adjuk meg egy osztály kitűnő tanulóit!
- Adjuk meg egy természetes szám összes osztóját!
- Adjuk meg egy mondat magas hangrendű szavait!
- Adjuk meg emberek egy halmazából a 180 cm felettieket!
- Adjuk meg egy év azon napjait, amikor délben nem fagyott!
- Soroljuk föl egy szó magánhangzóit!

8. Kiválogatás

Feladatok:

- Adjuk meg egy osztály kitűnő tanulóit!
- Adjuk meg egy természetes szám összes osztóját!
- Adjuk meg egy mondat magas hangrendű szavait!
- Adjuk meg emberek egy halmazából a 180 cm felettieket!
- Adjuk meg egy év azon napjait, amikor délben nem fagyott!
- Soroljuk föl egy szó magánhangzóit!

Mi bennük a közös?

N darab „valami” közül kell megadni az összes, adott T tulajdonsággal rendelkezőt!

8. Kiválogatás

N darab „valami” közül kell megadni az összes, adott T tulajdonsággal rendelkezőt!

Specifikáció:

- Bemenet: $N\in\mathbb{N}, X_{1..N}\in\mathbb{H}^N,$
  $T:\mathbb{H}\to\mathbb{L}$
- Kimenet: $Db\in\mathbb{N}, Y_{1..N}\in\mathbb{N}^N$
- Előfeltétel: –
- Utófeltétel: $Db=\sum_{\substack{i=1\\T(X_i)}}^{N} 1$ és
  $\forall i(1\le i\le Db): T(X_{Y_i})$ és
  $Y\subseteq(1,2,\dots,N)$

Az első Db elemet használva

L. Megszámolás tételt!

Másképp: $(Db,Y)=\underset{\substack{i=1\\T(X_i)}}{\overset{N}{\text{Kiválogat}}}\ i$

8. Kiválogatás

Algoritmus:

Specifikáció:

- Bemenet: $N \in \mathbb{N}, X_{1..N} \in H^N, T: H \to \mathbb{L}$
- Kimenet: $Db \in \mathbb{N}, Y_{1..N} \in \mathbb{N}^N$
- Előfeltétel: –
- Utófeltétel: $Db = \sum_{\substack{i=1 \\ T(X_i)}}^{N} 1$ és $\forall i (1 \le i \le Db): T(X_{Y_i})$ és $Y \subseteq (1,2,\ldots,N)$

L. Megszámolás tételt!

Változó
i:Egész

Db:=0

i=1..N

T(X[i])

I | N

Db:=Db+1 | —

Y[Db]:=i

Megjegyzés:

A sorszám általánosabb, mint az érték. Ha mégis érték kellene, akkor Y[Db]:=X[i] szerepelne. (Ekkor a specifikációt is módosítani kell! Lásd később!)

8. Kiválogatás

N darab „valami” közül kell megadni az összes, adott T tulajdonsággal rendelkezőt!

Értékek kiválogatása (tömören):

Specifikáció$_2$:

- Kimenet: $Db \in \mathbb{N}, Y_{1..N} \in H^N$
- Utófeltétel: $Db = \sum_{\substack{i=1 \\ T(X_i)}}^{N} 1$ és

  $\forall i(1 \le i \le Db): T(Y_i)$ és

  $Y \subseteq X$

Másképp: $(Db, Y) = \underset{\substack{i=1 \\ T(X_i)}}{\overset{N}{\text{Kiválogat}}}\ X_i$

Specifikáció:

- Bemenet: $N \in \mathbb{N}, X_{1..N} \in H^N,$
  $T: H \to L$
- Kimenet: $Db \in \mathbb{N}, Y_{1..N} \in \mathbb{N}^N$
- Előfeltétel: –
- Utófeltétel: $Db = \sum_{\substack{i=1 \\ T(X_i)}}^{N} 1$ és

  $\forall i(1 \le i \le Db): T(X_{Y_i})$ és

  $Y \subseteq (1,2,\ldots,N)$

8. Kiválogatás

Specifikáció: $T(X_i) \rightarrow H_i>0$

- Bemenet: $N \in \mathbb{N}$, $H_{1..N} \in \mathbb{R}^N$, $Poz: \mathbb{R} \rightarrow \mathbb{L}$, $Poz(x) := x>0$
- Kimenet: $Db \in \mathbb{N}$, $NF_{1..N} \in \mathbb{N}^N$
- Előfeltétel: –
- Utófeltétel$_1$: $Db = \sum_{\substack{i=1 \\ H_i>0}}^{N} 1$ és $\forall i(1 \leq i \leq Db)$: $H_{NF_i}>0$ és $NF \subseteq (1,2,\ldots,N)$
- Utófeltétel$_2$: $(Db, NF) = \underset{\substack{i=1 \\ H_i>0}}{\overset{N}{\text{Kiválogat}}}\, i$

- Adjuk meg egy év azon napjait, amikor délben nem fagyott!

Specifikáció:

- Bemenet: $N \in \mathbb{N}$, $X_{1..N} \in H^N$, $T: H \rightarrow \mathbb{L}$
- Kimenet: $Db \in \mathbb{N}$, $Y_{1..N} \in \mathbb{N}^N$
- Előfeltétel: –
- Utófeltétel: $Db = \sum_{\substack{i=1 \\ T(X_i)}}^{N} 1$ és $\forall i(1 \leq i \leq Db)$: $T(X_{Y_i})$ és $Y \subseteq (1,2,\ldots,N)$

8. Kiválogatás

Algoritmus: $T(X[i]) \rightarrow H[i]>0$

Változó
i:Egész

Db:=0

i=1..N

H[i]>0 (I / N)

Db:=Db+1

NF[Db]:=i

—

Specifikáció:

- Bemenet: $N\in\mathbb{N}$, $H\in\mathbb{R}^N$,
  $Poz:\mathbb{R}\rightarrow\mathbb{L}$, $Poz(x):=x>0$
- Kimenet: $Db\in\mathbb{N}$, $NF\in\mathbb{N}^{Db}$
- Előfeltétel: –
- Utófeltétel$_1$: $Db=\sum_{\substack{i=1\\H_i>0}}^{N} 1$ és
  $\forall i(1\le i\le Db): H_{NF_i}>0$ és
  $NF\subseteq(1,2,\ldots,N)$

Specifikáció:

- Bemenet: $N\in\mathbb{N}$, $X_{1..N}\in H^N$,
  $T:H\rightarrow\mathbb{L}$
- Kimenet: $Db\in\mathbb{N}$, $Y_{1..N}\in\mathbb{N}^N$
- Előfeltétel: –
- Utófeltétel: $Db=\sum_{\substack{i=1\\T(X_i)}}^{N} 1$ és
  $\forall i(1\le i\le Db): T(X_{Y_i})$ és
  $Y\subseteq(1,2,\ldots,N)$

Db:=0

i=1.. N

T(X[i]) (I / N)

Db:=Db+1

Y[Db]:=i

—

8. Kiválogatás helyben

Programparaméterek:

Konstans
MaxN:**Egész**(???)
Típus
THk=**Tömb**[1..MaxN:TH]
Változó
N:**Egész**, X:THk
…

Specifikáció:

- Bemenet: $N \in \mathbb{N},\ X_{1..N} \in \mathbb{H}^N$
- Kimenet: $Db \in \mathbb{N},\ Y_{1..N} \in \mathbb{H}^N$
- Előfeltétel: –
- Utófeltétel: $Db = \sum_{\substack{i=1 \\ T(X_i)}}^{N} 1$ és $Y_{1..Db} \subseteq X$ és $\forall i(1 \le i \le Db): T(Y_i)$

Itt a bemenetben szereplő X és a kimenetben szereplő Y lehet a programban ugyanaz a változó. Jelöljük ezt is X-szel. Teljesülni kell rá a megálláskor (meghagyva a specifikációbeli műveleteket):

$X^{kimeneti}_{1..Db} \subseteq X^{bemeneti}_{1..N}$ és $\forall i(1 \le i \le Db): T(X^{kimeneti}_i)$

8. Kiválogatás helyben

Ötlet:

Itt olyan helyre tesszük a kiválogatott elemet, amelyre már nincs szükségünk.

Algoritmus:

Specifikáció:

- Bemenet: $N \in \mathbb{N}$, $X \in H^N$
- Kimenet: $Db \in \mathbb{N}$, $X' \in H^N$
- Előfeltétel: –
- Utófeltétel: $Db = \sum_{\substack{i=1 \\ T(X_i)}}^{N} 1$ és
  $X'_{1..Db} \subseteq X$ és
  $\forall i(1 \le i \le Db): T(X'_i)$

Változó
i:**Egész**

Db:=0

i=1..N

T(X[i])

I: Db:=Db+1; X[Db]:=X[i]

N: —

Speciális sorozat típus: dinamikus tömb

A programozás a tömb típuson kívül sokféle sorozat típust ismer. Közülük az egyik egy olyan indexelhető típus, aminek az elemszáma futás közben növelhető (ebből a szempontból a szöveg típusra hasonlít).

Műveletei:

- Hossz(S) – az S sorozat és a neki megfelelő tömb elemei száma
- Végére(S,x) – az S tömb végére egy új elemet, az x-et illeszti
- S[i] – az S tömb i-edik eleme

További műveletek is lehetnek, most nem térünk ki rá.

Figyelem: e típus használata jelentősen megnövelheti a program futási idejét!

Speciális sorozat típus: dinamikus tömb C#-ban

Deklaráció:

S:Tömb[TElem**]** – `List<TElem> S`

… és az üres tömb létrehozása:

`new List<TElem>();`

Deklaráció + létrehozás:

`List<TElem> S = new List<TElem>();`

Műveletek:

- Hossz(S) – `S.Count`
- Végére(S,x) – `S.Add(x)`
- S[i] – `S[i]`

8. Kiválogatás dinamikus tömbbe

Specifikáció:

N darab „valami” közül kell megadni az összes, adott T tulajdonsággal rendelkezőt!

- Bemenet: $N \in \mathbb{N}$, $X_{1..N} \in \mathbb{H}^N$, $T:\mathbb{H} \to \mathbb{L}$
- Kimenet: $Y_{1..} \in \mathbb{N}^*$
- Előfeltétel: –
- Utófeltétel: $Hossz(Y) = \sum_{\substack{i=1 \\ T(X_i)}}^{N} 1$ és $\forall y \in Y$: $T(X_y)$ és $Y \subseteq (1,2,\ldots,N)$

Annyi elemet használva, amennyit kell.

Specifikáció:

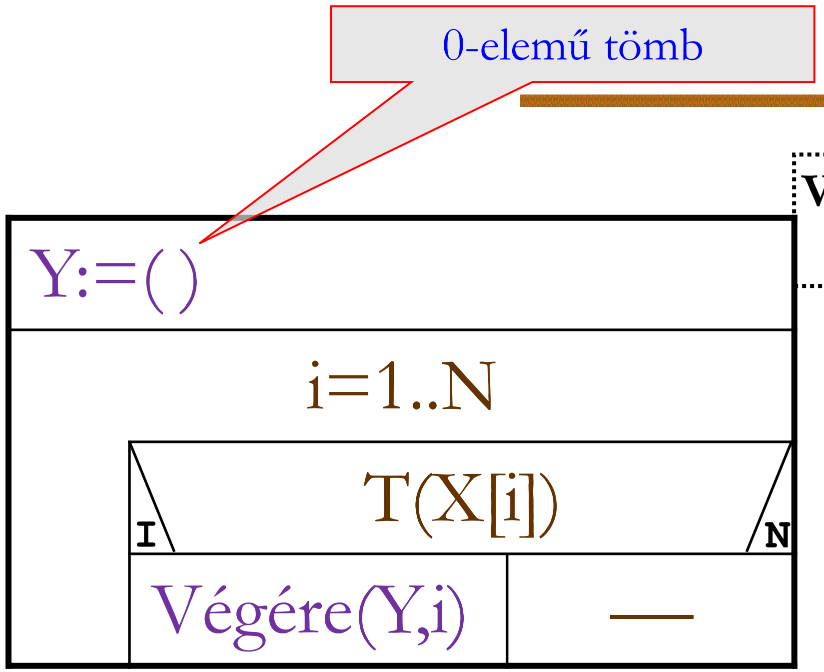
- Bemenet: $N \in \mathbb{N}$, $X_{1..N} \in \mathbb{H}^N$, $T:\mathbb{H} \to \mathbb{L}$
- Kimenet: $Db \in \mathbb{N}$, $Y_{1..N} \in \mathbb{N}^N$
- Előfeltétel: –
- Utófeltétel: $Db = \sum_{\substack{i=1 \\ T(X_i)}}^{N} 1$ és $\forall i(1 \le i \le Db)$: $T(X_{Y_i})$ és $Y \subseteq (1,2,\ldots,N)$

8. Kiválogatás dinamikus tömbbe

Algoritmus:

Specifikáció:

- Bemenet: $N \in \mathbb{N}$, $X_{1..N} \in H^N$, $T: H \to \mathbb{L}$
- Kimenet: $Y \in \mathbb{N}^*$
- Előfeltétel: –
- Utófeltétel: $\text{Hossz}(Y) = \sum_{\substack{i=1 \\ T(X_i)}}^{N} 1$ és $\forall y \in Y$: $T(X_y)$ és $Y \subseteq (1,2,\ldots,N)$



Változó
i:Egész

Megjegyzés:

A sorszám általánosabb, mint az érték. Ha mégis **érték** kellene, akkor Végére(Y,X[i]) szerepelne. (Ekkor a specifikációt is módosítani kell!)

10. Szétválogatás

Feladatok:

- Adjuk meg egy számsorozatból a páros és a páratlan számokat is!
- Adjuk meg egy év azon napjait, amikor délben fagyott és amikor nem fagyott!
- Adjuk meg egy angol szó magán- és mássalhangzóit!
- Adjuk meg emberek egy halmazából a 140 cm alattiakat, a 140 és 180 cm közöttieket és a 180 cm felettieket!
- Adjuk meg emberek egy halmazából a télen, tavasszal, nyáron, illetve ősszel születetteket!

10. Szétválogatás

Feladatok:

- Adjuk meg egy számsorozatból a páros és a páratlan számokat is!
- Adjuk meg egy év azon napjait, amikor délben fagyott és amikor nem fagyott!
- Adjuk meg egy angol szó magán- és mássalhangzóit!
- Adjuk meg emberek egy halmazából a 140 cm alattiakat, a 140 és 180 cm közöttieket és a 180 cm felettieket!
- Adjuk meg emberek egy halmazából a télen, tavasszal, nyáron, illetve ősszel születetteket!

Mi bennük a közös?

N darab „valami” közül kell megadni az összes, adott T tulajdonsággal rendelkezőt, illetve nem rendelkezőt! Azaz az összes bemeneti elemet „besoroljuk” a kimenet valamely sorozatába.

A többfelé szétválogatás visszavezethető a kétfelé szétválogatásra.

10. Szétválogatás

N darab „valami” közül kell megadni az összes, adott T tulajdonsággal rendelkezőt, illetve nem rendelkezőt!

Specifikáció:

- Bemenet: $N \in \mathbb{N}$, $X_{1..N} \in \mathbb{H}^N$, $T: \mathbb{H} \to \mathbb{L}$
- Kimenet: $Db \in \mathbb{N}$, $Y_{1..N} \in \mathbb{N}^N$, $Z_{1..N} \in \mathbb{N}^N$
- Előfeltétel: –
- Utófeltétel: $Db = \sum_{\substack{i=1 \\ T(X_i)}}^{N} 1$ és

  $\forall i(1 \leq i \leq Db): T(X_{Y_i})$ és

  $\forall i(1 \leq i \leq N-Db):$ nem $T(X_{Z_i})$ és

  $Y \subseteq (1,2,\ldots,N)$ és $Z \subseteq (1,2,\ldots,N)$

10. Szétválogatás

N darab „valami” közül kell megadni az összes, adott T tulajdonsággal rendelkezőt, illetve nem rendelkezőt!

Specifikáció$_2$:

- Utófeltétel$_2$: $(Db, Y, Z) = \underset{\substack{i=1 \\ T(X_i)}}{\overset{N}{\text{Szétválogat}}}\ i$

Értékek szétválogatása esetén:

$$(Db, Y, Z) = \underset{\substack{i=1 \\ T(X_i)}}{\overset{N}{\text{Szétválogat}}}\ X_i$$

10. Szétválogatás

Algoritmus:

Specifikáció:

- Bemenet: $N \in \mathbb{N}$
  $X_{1..N} \in H^N$
  $T: H \rightarrow \mathbb{L}$
- Kimenet: $Db \in \mathbb{N}$
  $Y_{1..N} \in \mathbb{N}^N$, $Z_{1..N} \in \mathbb{N}^N$
- Előfeltétel: –
- Utófeltétel: $Db = \sum_{\substack{i=1 \\ T(X_i)}}^{N} 1$ és
  $\forall i(1 \le i \le Db): T(X_{Y_i})$ és
  $\forall i(1 \le i \le N-Db)$: nem $T(X_{Z_i})$ és
  $Y \subseteq (1,2,\ldots,N)$ és $Z \subseteq (1,2,\ldots,N)$

Változó
i:Egész

Db:=0

i=1..N

T(X[i])

I	N
Db:=Db+1	Z[i-Db]:=i
Y[Db]:=i	

Megjegyzés:

Itt is szerepelhetne :=i helyett :=X[i], ha csak az értékekre lenne szükségünk. (A specifikáció is módosítandó!)

10. Szétválogatás

Probléma:

Y-ban és Z-ben együtt csak N darab elem van, azaz elég lenne egyetlen N-elemű sorozat.

Megoldás:

- Bemenet: $N \in \mathbb{N}$, $X_{1..N} \in H^N$, $T:H \rightarrow \mathbb{L}$
- Kimenet: $Db \in \mathbb{N}$, $Y_{1..N} \in \mathbb{N}^N$
- Előfeltétel: –
- Utófeltétel: $Db = \sum_{\substack{i=1 \\ T(X_i)}}^{N} 1$ és

  $\forall i(1 \leq i \leq Db): T(X_{Y_i})$ és

  $\forall i(Db+1 \leq i \leq N)$: nem $T(X_{Y_i})$ és

  $Y \in Permutáció(1,2,\ldots,N)$

Specifikáció:

- Bemenet: $N \in \mathbb{N}$
  $X_{1..N} \in H^N$
  $T:H \rightarrow \mathbb{L}$
- Kimenet: $Db \in \mathbb{N}$
  $Y_{1..N} \in \mathbb{N}^N$, $Z_{1..N} \in \mathbb{N}^N$
- Előfeltétel: –
- Utófeltétel: $Db = \sum_{\substack{i=1 \\ T(X_i)}}^{N} 1$ és
  $\forall i(1 \leq i \leq Db): T(X_{Y_i})$ és
  $\forall i(1 \leq i \leq N-Db)$: nem $T(X_{Z_i})$ és
  $Y \subseteq (1,2,\ldots,N)$ és $Z \subseteq (1,2,\ldots,N)$

10. Szétválogatás

N darab „valami” közül kell megadni az összes, adott T tulajdonsággal rendelkezőt, illetve nem rendelkezőt!

Specifikáció$_2$:

- Utófeltétel$_2$: $(Db, Y) = \underset{\substack{i=1\\T(X_i)}}{\overset{N}{\text{Szétválogat}_2}}\, i$

Értékek szétválogatása esetén:

$$(Db, Y) = \underset{\substack{i=1\\T(X_i)}}{\overset{N}{\text{Szétválogat}_2}}\, X_i$$

10. Szétválogatás

Algoritmus:

- Bemenet: $N \in \mathbb{N}$, $X_{1..N} \in H^N$
- Kimenet: $Db \in \mathbb{N}$, $Y_{1..N} \in \mathbb{N}^N$
- Előfeltétel: –
- Utófeltétel: $Db = \sum_{\substack{i=1 \\ T(X_i)}}^{N} 1$ és
  $\forall i(1 \le i \le Db): T(X_{Y_i})$ és
  $\forall i(Db+1 \le i \le N): \text{nem } T(X_{Y_i})$ és
  $Y \in \text{Permutáció}(1,2,\ldots,N)$

<table>
<tr><td colspan="3">Db:=0 [≅elölről index]</td></tr>
<tr><td colspan="3">ind2:=N+1 [≅hátulról index]</td></tr>
<tr><td colspan="3">i=1..N</td></tr>
<tr><td rowspan="3"></td><td colspan="2">T(X[i])
I | N</td></tr>
<tr><td>Db:=Db+1</td><td>ind2:=ind2–1</td></tr>
<tr><td>Y[Db]:=i</td><td>Y[ind2]:=i</td></tr>
</table>

Változó
ind2,
i:**Egész**

Megjegyzés: Itt célszerű egy segédváltozó arra, hogy hol tartunk Y-ban hátulról: ind2.

10. Szétválogatás dinamikus tömbökbe

A kiválogatáshoz hasonlóan itt is használhatunk az eredmények tárolásához bővíthető elemszámú sorozatokat.

N darab „valami” közül kell megadni az összes, adott T tulajdonsággal rendelkezőt, illetve nem rendelkezőt!

Specifikáció:

- Bemenet: $N \in \mathbb{N}, X_{1..N} \in H^N, T:H \to \mathbb{L}$
- Kimenet: $Y_{1..} \in \mathbb{N}^*, Z_{1..} \in \mathbb{N}^*$
- Előfeltétel: –
- Utófeltétel: $hossz(Y)=\sum_{\substack{i=1 \\ T(X_i)}}^{N} 1$ és $Y \subseteq (1,2,\ldots,N)$ és $\forall y \in Y: T(X_y)$ és
  $hossz(Z)=\sum_{\substack{i=1 \\ \text{nem } T(X_i)}}^{N} 1$ és $Z \subseteq (1,2,\ldots,N)$ és $\forall z \in Z: \text{nem } T(X_z)$

10. Szétválogatás dinamikus tömbökbe

Algoritmus:

Specifikáció:

- Bemenet: $N \in \mathbb{N}, X_{1..N} \in H^N, T: H \to \mathbb{L}$
- Kimenet: $Y \in \mathbb{N}^*, Z \in \mathbb{N}^*$
- Előfeltétel: –
- Utófeltétel: $hossz(Y) = \sum_{\substack{i=1 \\ T(X_i)}}^{N} 1$ és $Y \subseteq (1,2,\ldots,N)$ és $\forall y \in Y: T(X_y)$ és

  $hossz(Z) = \sum_{\substack{i=1 \\ nem\ T(X_i)}}^{N} 1$ és $Z \subseteq (1,2,\ldots,N)$ és $\forall z \in Z: nem\ T(X_z)$

Változó
i:Egész

Y:=( ); Z:=( )

i=1..N

T(X[i])

I	N
Végére(Y,i)	Végére(Z,i)

10. Szétválogatás helyben

Programparaméterek:

Konstans
MaxN:**Egész**(???)
Típus
THk=**Tömb**[1..MaxN:TH]
Változó
N:**Egész**, X:THk
…

Specifikáció:

- Bemenet: $N \in \mathbb{N},\ X_{1..N} \in \mathbb{H}^N$
- Kimenet: $Db \in \mathbb{N},\ Y_{1..N} \in \mathbb{H}^N$
- Előfeltétel: –
- Utófeltétel: $Db = \sum_{\substack{i=1 \\ T(X_i)}}^{N} 1$ és $Y \in Permutáció(X)$ és

$$\forall i(1 \le i \le Db): T(Y_i) \text{ és}$$
$$\forall i(Db+1 \le i \le N): \text{nem } T(Y_i)$$

Megjegyzés: bemenetben szereplő X és a kimenetben szereplő Y legyen a programban ugyanaz az X változó!

10. Szétválogatás helyben

Algoritmikus ötlet:

1. Vegyük ki (másoljuk le) a sorozat **első** elemét:

   O x x x x x x x x x x x x

2. Keresünk **hátulról** egy elemet, aminek **elől** a helye (mert T tulajdonságú, nem odavaló):

   O x x x x x x **x** x x x x x

3. A megtalált elemet **tegyük** az előbb keletkezett **lyukba**:

   ⊗ x x x x x x O x x x x x

A lyuk mögött és az 1. elemmel már rendben vagyunk.

10. Szétválogatás helyben

4. Most keletkezett egy lyuk hátul. Az előbb betöltött lyuktól indulva elölről keressünk hátra teendő (nem odavaló: nem T-tulajdonságú) elemet:

⊗ x x x x x O x x x x x

5. A megtalált elemet tegyük a hátul levő lyukba, majd újra hátulról kereshetünk!

⊗ x x O x x ⊗ x x x x x

Az elől keletkezett lyuk előttiek és a hátrébb mozgatott elemmel kezdve rendben vagyunk.

10. Szétválogatás helyben

6. … és így tovább …

7. Befejezzük a keresést, ha valahonnan elértük a lyukat.

   x x x x O x x x x x x x x

8. Erre a helyre az 1. lépésben kivettet visszatesszük.

Utófeltétel pontosítása:

Teljesülni kell az X vektorra a megálláskor (meghagyva a specifiká-cióbeli műveleteket):

$X_{1..N}^{kimeneti}$=permutáció($X_{1..N}^{bemeneti}$) és

$\forall i(1 \le i \le Db)$: T($X_i^{kimeneti}$) és $\forall i(Db+1 \le i \le N)$: nem T($X_i^{kimeneti}$)

10. Szétválogatás helyben

Algoritmus:

Specifikáció:
- Bemenet: $N \in \mathbb{N}$, $X_{1..N} \in H^N$
- Kimenet: $Db \in \mathbb{N}$, $X'_{1..N} \in H^N$
- Előfeltétel: –
- Utófeltétel: $Db = \sum_{\substack{i=1 \\ T(X_i)}}^{N} 1$ és $X' \in Permutáció(X)$

  és $\forall i(1 \le i \le Db)$: $T(X'_i)$
  és $\forall i(Db+1 \le i \le N)$: nem $T(X'_i)$

Változó
- e,u:**Egész**
- y:TH
- Van:**Logikai**

- e:=1 [a szétválogatandók elsője]
- u:=N [a szétválogatandók utolsója]
- y:=X[e]
- ciklus amíg e<u
  - HátulrólKeres(e,u,Van)
  - Van?
    - I:
      - X[e]:=X[u]
      - e:=e+1
      - ElölrőlKeres(e,u,Van)
      - Van?
        - I:
          - X[u]:=X[e]
          - u:=u–1
        - N: —
    - N: —
- …

10. Szétválogatás helyben

Algoritmus:

Specifikáció:

- Bemenet: $N \in \mathbb{N}$, $X_{1..N} \in H^N$
- Kimenet: $Db \in \mathbb{N}$, $X'_{1..N} \in H^N$
- Előfeltétel: –
- Utófeltétel: $Db = \sum_{\substack{i=1 \\ T(X_i)}}^{N} 1$ és $X' \in Permutáció(X)$
  és $\forall i(1 \leq i \leq Db): T(X'_i)$
  és $\forall i(Db+1 \leq i \leq N)$: nem $T(X'_i)$

…	
X[e]:=y	
I — T(y) — N	
Db:=e	Db:=e–1

Megjegyzés: Az X változóról az algoritmus végrehajtása közben különböző állításokat mondhatunk:

1. kezdetben a bemenetbeli sorozat;
2. a futás végén a bemeneti X permutációja a szétválogatás utófeltétele szerint;
3. közben e-ig T tulajdonságú elemek, u-tól nem T tulajdonságú elemek, köztük nem vizsgált elemek.

Ún. ciklusinvariáns

10. Szétválogatás helyben

ElölrőlKeres(e,u:**Egész**,Van:**Logikai**)

e<u és T(X[e])
e:=e+1
Van:=e<u

HátulrólKeres(e,u:**Egész**,Van:**Logikai**)

e<u és nem T(X[u])
u:=u–1
Van:=e<u

Programozási tételek

- **Sorozat → sorozat**

7. Másolás – függvényszámítás
8. Kiválogatás
9. Rendezés (később lesz)

- **Sorozat → sorozatok**

10. Szétválogatás

Mátrixok

Feladat:

Egy N×M-es raszterképet nagyítsunk a kétszeresére pontsokszorozással: minden régi pont helyébe 2×2 azonos színű pontot rajzolunk a nagyított képen.

→

Mátrixok

Problémák/**válaszok**:

- Hogyan ábrázoljunk egy képet?

  A kép rendezett pontokból áll, azaz biztosan valamilyen sorozatként adható meg.

- Nehézkes lenne azonban a pontokra egy sorszámozást adni.

  Kézenfekvőbb azt megmondani, hogy egy képpont a kép hányadik sorában, illetve oszlopában található, azaz alkalmazzunk **dupla indexelés**t!

  A kétindexes tömböket hívjuk **mátrix**nak.

Mátrixok

Specifikáció:

- Bemenet: $N,M \in \mathbb{N}$, $K_{1..N,1..M} \in \mathbb{N}^{N \times M}$
- Kimenet: $NK_{1..2*N,1..2*M} \in \mathbb{N}^{2*N \times 2*M}$
- Előfeltétel: –
- Utófeltétel: $\forall i(1 \le i \le N)$: $\forall j(1 \le j \le M)$:
  $NK_{2*i,2*j} = K_{i,j}$ és
  $NK_{2*i-1,2*j} = K_{i,j}$ és
  $NK_{2*i,2*j-1} = K_{i,j}$ és
  $NK_{2*i-1,2*j-1} = K_{i,j}$

Feladat:

Egy N*M-es raszterképet nagyítsunk a kétszeresére *pontsokszorozás*sal: minden régi pont helyébe 2*2 azonos színű pontot rajzolunk a nagyított képen.

$:=(\mathbb{N}^{M})^{N}$

N – a sorok,

M – az oszlopok száma

Ez a **másolás** tétel egy variációja, csak egy elemből négy elem keletkezik.

Mátrixok

Algoritmus – adatleírás:

Specifikáció:

- Bemenet: $N,M \in \mathbb{N}$, $K_{1..N,1..M} \in \mathbb{N}^{N \times M}$
- Kimenet: $NK_{1..2*N,1..2*M} \in \mathbb{N}^{2 \cdot N \times 2 \cdot M}$
- Előfeltétel: –
- Utófeltétel: $\forall i(1 \leq i \leq N)$: $\forall j(1 \leq j \leq M)$:
  $NK_{2 \cdot i,2 \cdot j}=K_{i,j}$ és
  $NK_{2 \cdot i-1,2 \cdot j}=K_{i,j}$ és
  $NK_{2 \cdot i,2 \cdot j-1}=K_{i,j}$ és
  $NK_{2 \cdot i-1,2 \cdot j-1}=K_{i,j}$

Konstans

MaxN:**Egész**(???)

MaxM:**Egész**(???)

Típus

TMátrix=**Tömb**[1..MaxN,1..MaxM:**Egész**]

Változó

N,M:**Egész**

K,NK:TMátrix

Mátrixok

Algoritmus:

Specifikáció:

- Bemenet: $N,M \in \mathbb{N}$, $K_{1..N,1..M} \in \mathbb{N}^{N \times M}$
- Kimenet: $NK_{1..2*N,1..2*M} \in \mathbb{N}^{2\cdot N \times 2\cdot M}$
- Előfeltétel: –
- Utófeltétel: $\forall i(1 \leq i \leq N)$: $\forall j(1 \leq j \leq M)$:
  $NK_{2\cdot i,2\cdot j}=K_{i,j}$ és
  $NK_{2\cdot i-1,2\cdot j}=K_{i,j}$ és
  $NK_{2\cdot i,2\cdot j-1}=K_{i,j}$ és
  $NK_{2\cdot i-1,2\cdot j-1}=K_{i,j}$

Változó
i,j:**Egész**

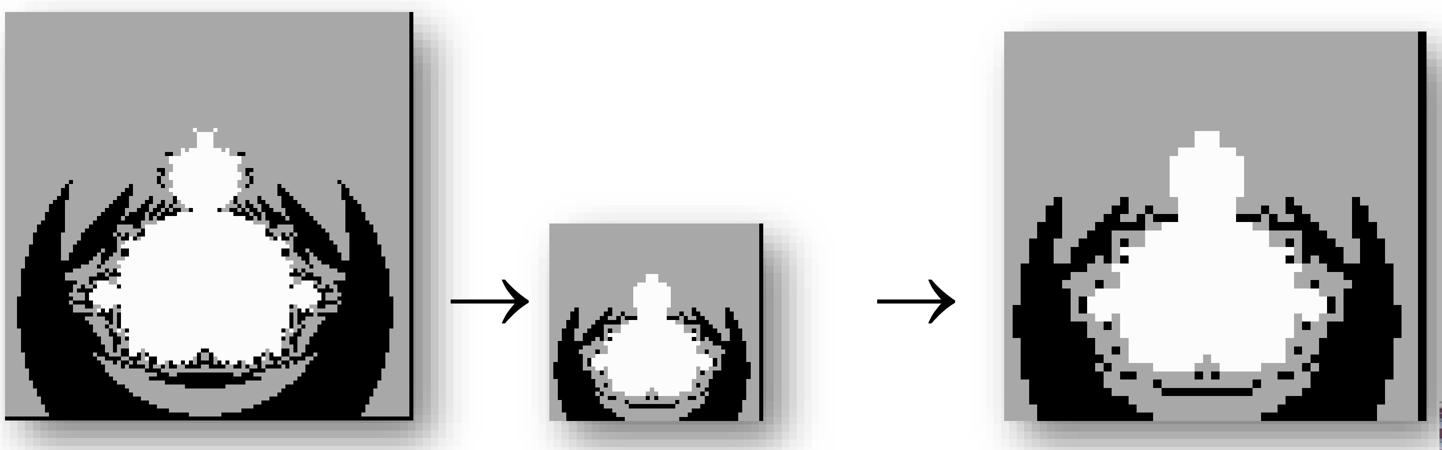
```
i=1..N
    j=1..M
        NK[2*i,2*j]:=K[i,j]
        NK[2*i–1,2*j]:=K[i,j]
        NK[2*i,2*j–1]:=K[i,j]
        NK[2*i–1,2*j–1]:=K[i,j]
```

Mátrixok

Feladat:

Egy N×M-es raszterképet kicsinyítsünk a felére (N/2×M/2 méretűre) pontátlagolással: a kicsinyített kép minden pontja az eredeti kép 2×2 pontjának „átlaga” legyen!

„átlag”: színkódok átlaga



Mátrixok

Specifikáció: (másolás)

- Bemenet: $N,M \in \mathbb{N}$, $K_{1..N,1..M} \in \mathbb{N}^{N \times M}$
- Kimenet: $KK_{1..N/2,1..M/2} \in \mathbb{N}^{N/2 \times M/2}$
- Előfeltétel: PárosE(N) és PárosE(M)
- Utófeltétel: $\forall i(1 \le i \le N/2): \forall j(1 \le j \le M/2):$

  $KK_{i,j}=(K_{2*i,2*j}+K_{2*i-1,2*j}+K_{2*i,2*j-1}+K_{2*i-1,2*j-1})$ Div 4
- Definíció: PárosE: $\mathbb{N} \rightarrow \mathbb{L}$

  PárosE(x):=(x Mod 2)=0

Feladat:

Egy N*M-es raszterképet kicsinyítsünk a felére (N/2*M/2 méretűre) *pontátlagolás*sal: a kicsinyített kép minden pontja az eredeti kép 2*2 pontjának „átlaga" legyen!

Mátrixok

Algoritmus:

➢ Utófeltétel: $\forall i(1 \leq i \leq N/2)$: $\forall j(1 \leq j \leq M/2)$:
$KK_{i,j}=(K_{2*i,2*j}+K_{2*i-1,2*j}+K_{2*i,2*j-1}+K_{2*i-1,2*j-1})$ Div 4

Változó
i,j:**Egész**

i=1..N/2
j=1..M/2
KK[i,j]:=(K[2*i,2*j]+K[2*i–1,2*j]+
K[2*i,2*j–1]+K[2*i–1,2*j–1]) Div 4

Megjegyzés:

1) a színes képeknél az átlagolással baj lehet! Milyen szín egy piros és egy kék színű pont átlaga? (hamis színek)
2) **RGB** esetén a szín: **Rekord**(piros,zöld,kék:**Egész**); és az átlag? (komponensenkénti átlag)

Mátrixok

Feladat:

A Rák-köd képére alkalmazzunk egyféle Rank-szűrőt! Minden pontot helyettesítsünk magának és a 8 szomszédjának maximumával!

→

Mátrixok

Specifikáció: (másolás+maximum-kiválasztás)

- Bemenet: $N,M\in\mathbb{N}$, $K_{1..N,1..M}\in\mathbb{N}^{N\times M}$
- Kimenet: $RK_{1..N,1..M}\in\mathbb{N}^{N\times M}$
- Előfeltétel: –
- Utófeltétel: $\forall i(1<i<N)$: $\forall j(1<j<M)$:

$$RK_{i,j}=\max_{p=i-1}^{i+1}\max_{q=j-1}^{j+1}K_{p,q} \text{ és}$$

$$\forall j(1\leq j\leq M): RK_{1,j}=K_{1,j} \text{ és } RK_{N,j}=K_{N,j}$$

$$\forall i(1\leq i\leq N): RK_{i,1}=K_{i,1} \text{ és } RK_{i,M}=K_{i,M}$$

Feladat:

A Rák-köd képére alkalmazzunk egyféle *Rank-szűrőt*! Minden pontot helyettesítsünk magának és a 8 szomszédjának maximumával!

Mátrixok

Algoritmus:

- Utófeltétel: $\forall i(1<i<N)$: $\forall j(1<j<M)$:

  $RK_{i,j}=\max\limits_{p=i-1}^{i+1}\max\limits_{q=j-1}^{j+1} K_{p,q}$ és

  $\forall i\ (1\le i\le N)$: $\forall j\ (1\le j\le M)$:

  $RK_{1,j}=K_{1,j}$ és $RK_{N,j}=K_{N,j}$

  $RK_{i,1}=K_{i,1}$ és $RK_{i,M}=K_{i,M}$

i=2..N–1

j=2..M–1

max:=0

p=i–1..i+1

q=j–1..j+1

K[p,q]>max

I | N

max:=K[p,q] | —

max:=…(i,j) körüli 3×3-as részmátrix maximuma…

RK[i,j]:=max

…

Változó
max,
i,j:**Egész**

Mátrixok

Algoritmus:

- Utófeltétel: $\forall i(1<i<N): \forall j(1<j<M):$

  $RK_{i,j}=\max\limits_{p=i-1}^{i+1}\max\limits_{q=j-1}^{j+1} K_{p,q}$ és

  $\forall i\ (1\le i\le N): \forall j\ (1\le j\le M):$

  $RK_{1,j}=K_{1,j}$ és $RK_{N,j}=K_{N,j}$

  $RK_{i,1}=K_{i,1}$ és $RK_{i,M}=K_{i,M}$

Specifikáció:

- Bemenet: $N\in\mathbb{N}$, $X\in H^N$
- Kimenet: $Max\in\mathbb{N}$
- Előfeltétel: $N>0$
- Utófeltétel: $1\le Max\le N$ és $\forall i\ (1\le i\le N): X_{Max}\ge X_i$

Max:=X[1]

i=2..N

X[i]>Max (I / N)

Max:=X[i] | —

Maximumérték-kiválasztás tétel.

i=2..N–1

j=2..M–1

max:=0

p=i–1..i+1

q=j–1..j+1

K[p,q]>max (I / N)

max:=K[p,q] | —

RK[i,j]:=max

. . .

Változó

max,

i,j:**Egész**

Mátrixok

Algoritmus (folytatás):

- Utófeltétel: $\forall i(1<i<N): \forall j(1<j<M):$

  $RK_{i,j} = \max\limits_{p=i-1}^{i+1} \max\limits_{q=j-1}^{j+1} K_{p,q}$ és

  $\forall j(1\leq j\leq M):$

  $RK_{1,j}=K_{1,j}$ és $RK_{N,j}=K_{N,j}$

  $\forall i(1\leq i\leq N):$

  $RK_{i,1}=K_{i,1}$ és $RK_{i,M}=K_{i,M}$

…
j=1..M
RK[1,j]:=K[1,j]
RK[N,j]:=K[N,j]
i=1..N
RK[i,1]:=K[i,1]
RK[i,M]:=K[i,M]

Változó
i,j:**Egész**