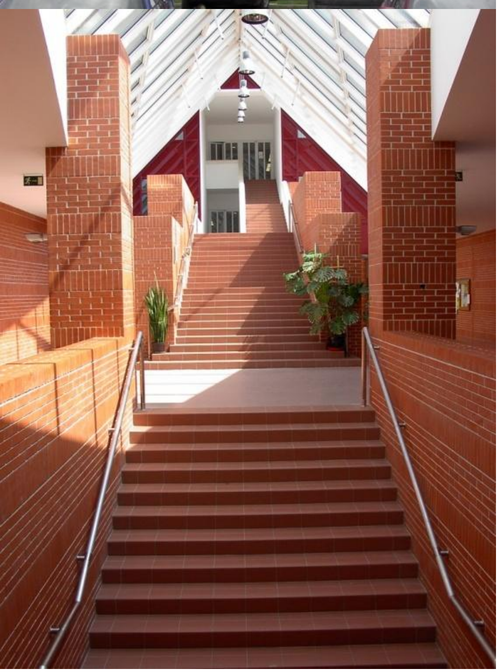
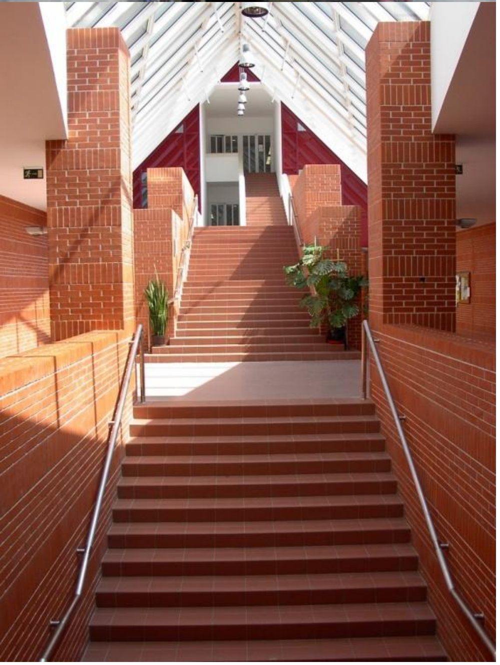
Programozási alapismeretek
10. előadás

Tartalom

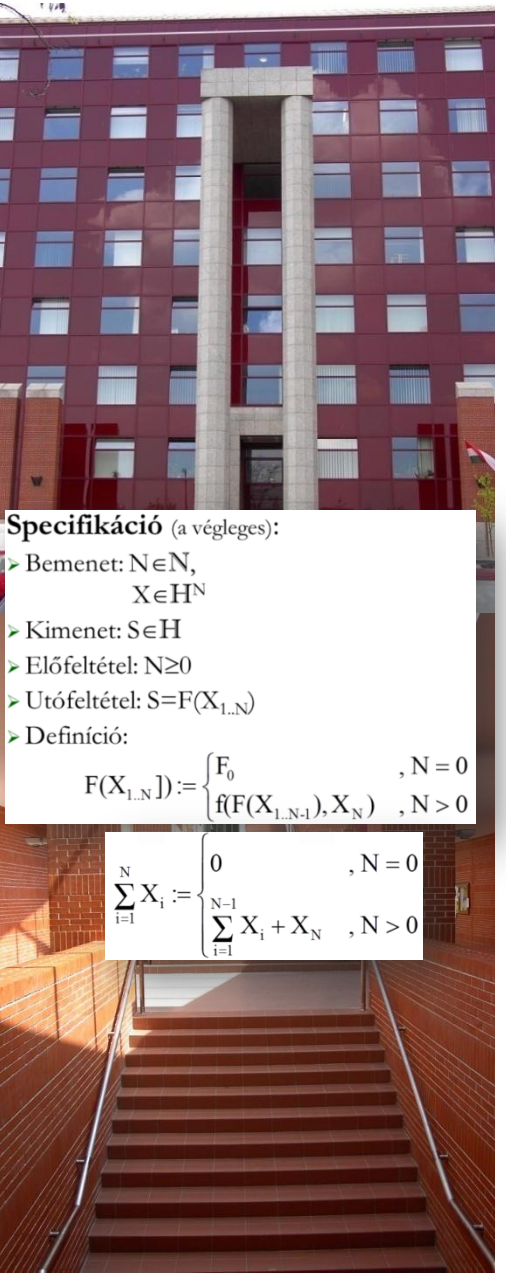


Kiválogatás + összegzés

Feladat:

Adott tulajdonságú elemek összege – **feltételes összegzés**.

Kiválogatás + összegzés

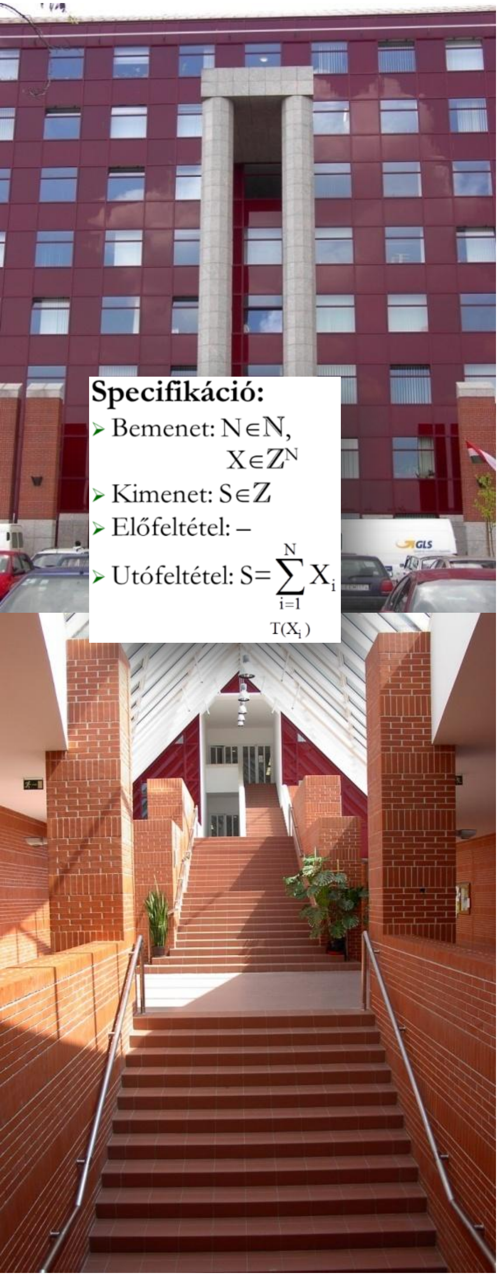


Feladat:

Adott tulajdonságú elemek összege – **feltételes összegzés**.

Specifikáció:

- Bemenet: $N \in \mathbb{N}$, $X \in \mathbb{Z}^N$
- Kimenet: $S \in \mathbb{Z}$
- Előfeltétel: –
- Utófeltétel: $S = \sum_{\substack{i=1 \\ T(X_i)}}^{N} X_i$

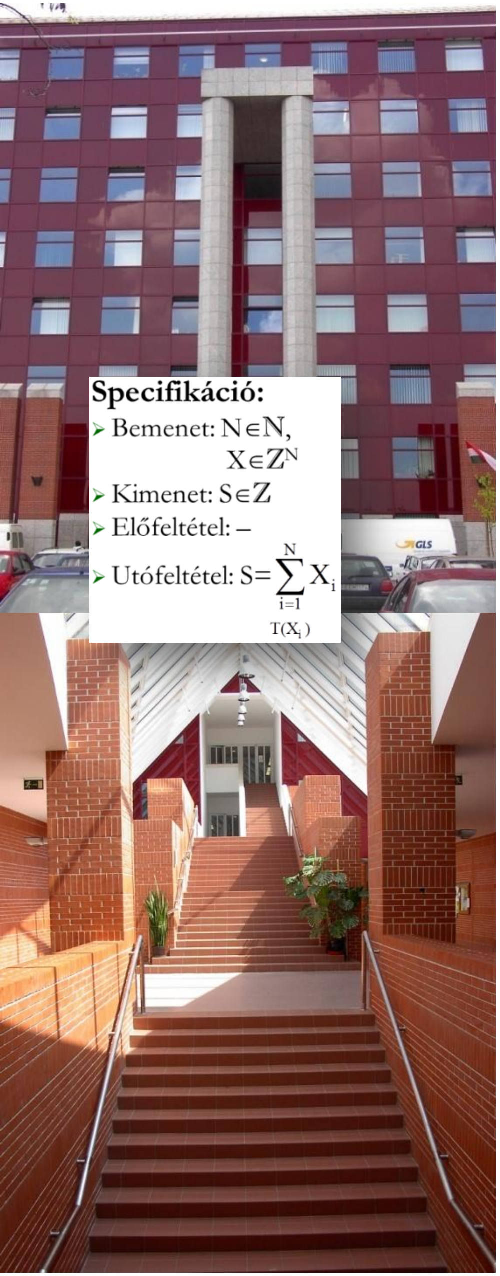


Kiválogatás + összegzés

Specifikáció$_a$:

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

$$S=\sum_{i=1}^{Db} X_{Y_i}$$



Kiválogatás + összegzés

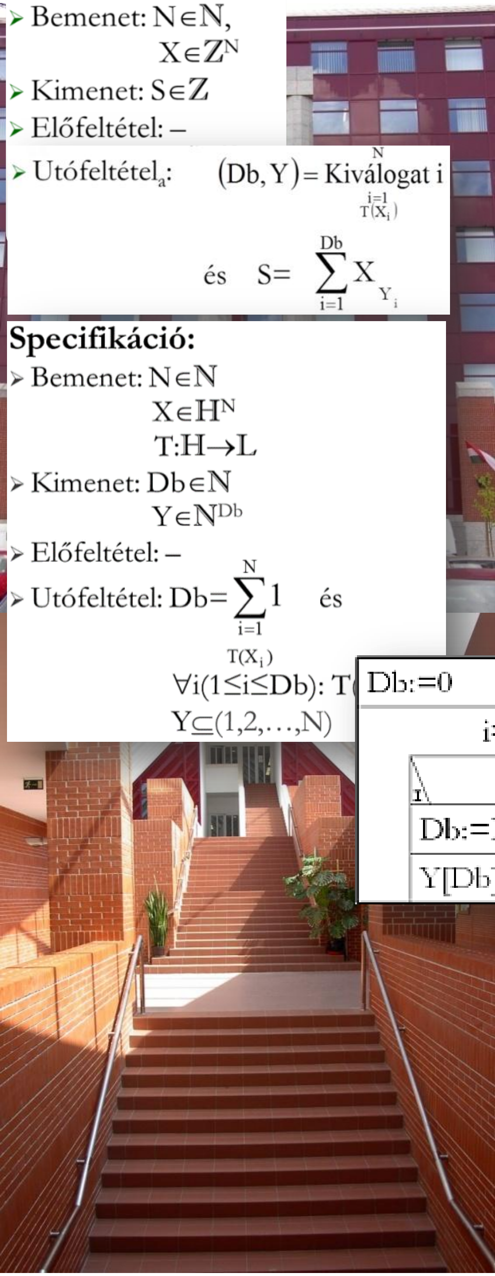
Specifikáció$_a$:

- Utófeltétel$_a$: $(Db,Y) = \underset{\substack{i=1 \\ T(X_i)}}{\overset{N}{\text{Kiválogat}}}\ i$ és $S = \sum_{i=1}^{Db} X_{Y_i}$

Specifikáció$_b$:

- Utófeltétel$_b$: $(Db,Y) = \underset{\substack{i=1 \\ T(X_i)}}{\overset{N}{\text{Kiválogat}}}\ X_i$ és $S = \sum_{i=1}^{Db} Y_i$

Kiválogatás + összegzés



Specifikáció:
- Bemenet: $N \in N$,
  $X \in Z^N$
- Kimenet: $S \in Z$
- Előfeltétel: –
- Utófeltétel$_a$: $(Db, Y) = \underset{\substack{i=1 \\ T(X_i)}}{\overset{N}{\text{Kiválogat}}}\, i$

  és $S = \sum_{i=1}^{Db} X_{Y_i}$

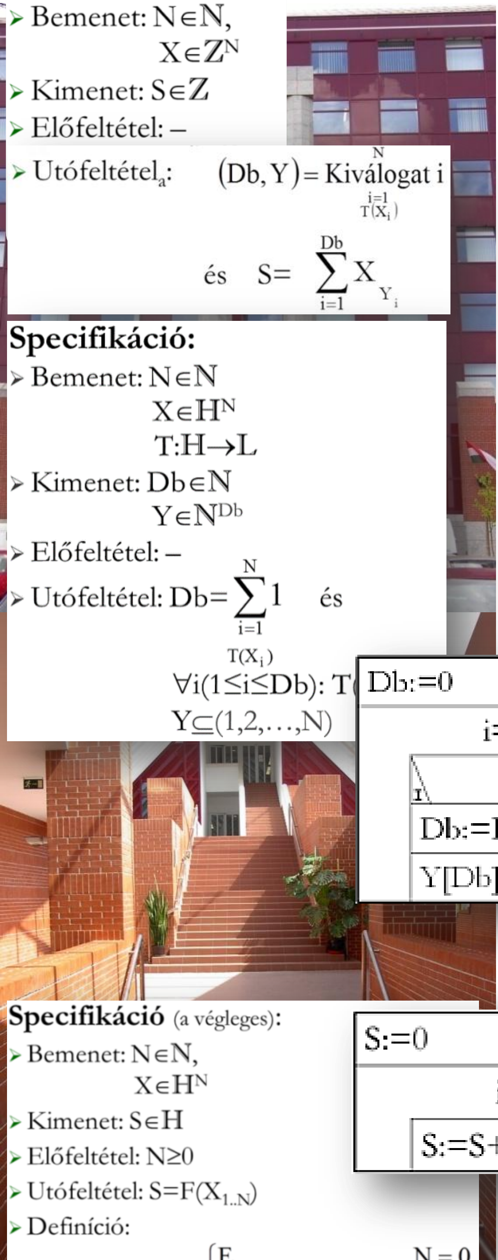
Specifikáció:
- Bemenet: $N \in N$
  $X \in H^N$
  $T:H \to L$
- Kimenet: $Db \in N$
  $Y \in N^{Db}$
- Előfeltétel: –
- Utófeltétel: $Db = \sum_{\substack{i=1 \\ T(X_i)}}^{N} 1$ és
  $\forall i(1 \le i \le Db)$: T(
  $Y \subseteq (1,2,\ldots,N)$

Db:=0	
i=1.. N	
T(X[i]) — I / N	
Db:=Db+1 Y[Db]:=i	—

1. megoldási ötlet$_a$:

Válogassuk ki az adott tulajdonságúakat,

Kiválogatás + összegzés



Specifikáció:
- Bemenet: $N \in \mathbb{N}$, $X \in \mathbb{Z}^N$
- Kimenet: $S \in \mathbb{Z}$
- Előfeltétel: –
- Utófeltétel$_a$: $(Db, Y) = \underset{\substack{i=1 \\ T(X_i)}}{\overset{N}{\text{Kiválogat}}}\, i$ és $S = \sum_{i=1}^{Db} X_{Y_i}$

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

```
Db:=0
i=1..N
    T(X[i])
    I                 N
    Db:=Db+1    |  —
    Y[Db]:=i    |
```

Specifikáció (a végleges):
- Bemenet: $N \in \mathbb{N}$, $X \in H^N$
- Kimenet: $S \in H$
- Előfeltétel: $N \ge 0$
- Utófeltétel: $S = F(X_{1..N})$
- Definíció:

$$F(X_{1..N}]) := \begin{cases} F_0 & , N = 0 \\ f(F(X_{1..N-1}), X_N) & , N > 0 \end{cases}$$

```
S:=0
i=1..N
    S:=S+X[i]
```

1. megoldási ötlet$_a$:

Válogassuk ki az adott tulajdonságúakat, majd utána adjuk össze őket!

Kiválogatás + összegzés

Specifikáció:

- Bemenet: $N \in \mathbb{N}$, $X \in \mathbb{Z}^N$
- Kimenet: $S \in \mathbb{Z}$
- Előfeltétel: –
- Utófeltétel$_a$: $(Db, Y) = \underset{\substack{i=1\\T(X_i)}}{\overset{N}{\text{Kiválogat}}}\, i$ és $S = \sum_{i=1}^{Db} X_{Y_i}$

Specifikáció:

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

Db:=0	
i=1.. N	
T(X[i]) (I / N)	
Db:=Db+1	—
Y[Db]:=i	

Specifikáció (a végleges):

- Bemenet: $N \in \mathbb{N}$, $X \in H^N$
- Kimenet: $S \in H$
- Előfeltétel: $N \ge 0$
- Utófeltétel: $S = F(X_{1..N})$
- Definíció:

$$F(X_{1..N}) := \begin{cases} F_0 & , N = 0 \\ f(F(X_{1..N-1}), X_N) & , N > 0 \end{cases}$$

S:=0
i=1..N
S:=S+X[i]

1. megoldási ötlet$_a$:

Válogassuk ki az adott tulajdonságúakat, majd utána adjuk össze őket!

Db:=0	
i=1..N	
T(X[i]) (I / N)	
Db:=Db+1	—
Y[Db]:=i	
S:=0	
i=1..Db	
S:=S+X[Y[i]]	

Változó

i,Db:**Egész**

Y:Tömb[…]

Kiválogatás + összegzés

Specifikáció:
- Bemenet: $N \in N$, $X \in Z^N$
- Kimenet: $S \in Z$
- Előfeltétel: –
- Utófeltétel$_b$: $(Db, Y) = \underset{\substack{i=1 \\ T(X_i)}}{\overset{N}{\text{Kiválogat}}}\ X_i$ és $S = \sum_{i=1}^{Db} Y_i$

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

Db:=0	
i=1.. N	
T(X[i]) I	N
Db:=Db+1	—
Y[Db]:=i	

Specifikáció (a végleges):
- Bemenet: $N \in N$, $X \in H^N$
- Kimenet: $S \in H$
- Előfeltétel: $N \ge 0$
- Utófeltétel: $S = F(X_{1..N})$
- Definíció:

$$F(X_{1..N}) := \begin{cases} F_0 & , N = 0 \\ f(F(X_{1..N-1}), X_N) & , N > 0 \end{cases}$$

S:=0
i=1..N
S:=S+X[i]

1. megoldási ötlet$_b$:

Válogassuk ki az adott tulajdonságúakat, majd utána adjuk össze őket!

Db:=0	
i=1..N	
T(X[i]) I	N
Db:=Db+1	—
Y[Db]:=X[i]	
S:=0	
i=1..Db	
S:=S+Y[i]	

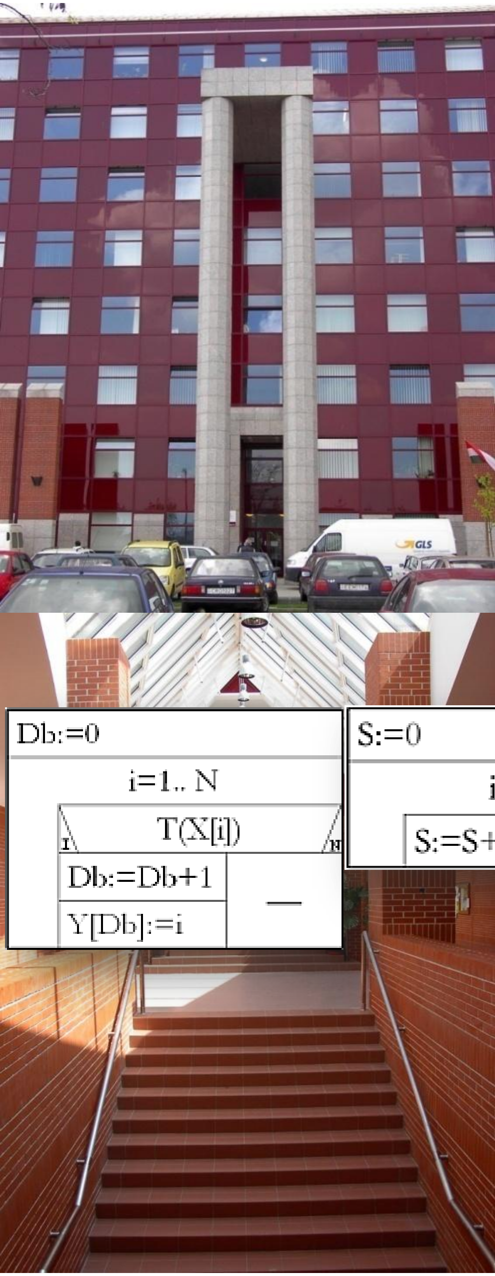
Változó
i,Db:**Egész**
Y:Tömb[…]

Kiválogatás + összegzés

2. megoldási ötlet:

Kiválogatás helyett azonnal adjuk össze a megfelelő elemeket! → nincs elem-/index-feljegyzés (Y-ban) + nincs számlálás (Db-ben)

Kiválogatás + összegzés

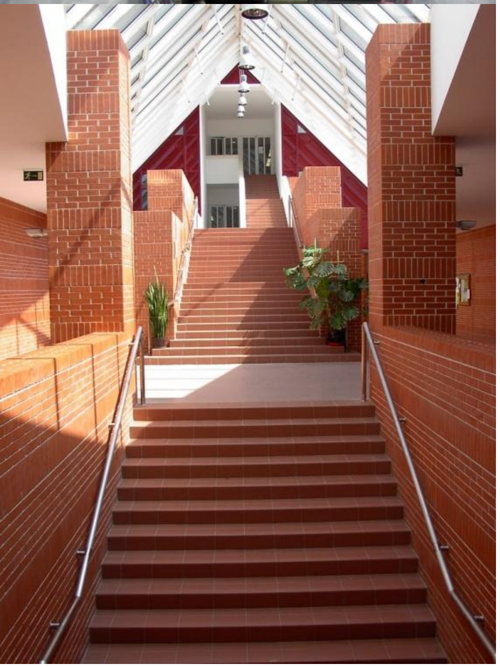


2. megoldási ötlet:

Kiválogatás helyett azonnal adjuk össze a megfelelő elemeket! → nincs elem-/index-feljegyzés (Y-ban) + nincs számlálás (Db-ben)

S:=0
i=1..N
T(X[i]) — I / N
I: S:=S+X[i]
N: —

Változó
i:**Egész**

Kiválogatás + maximum-kiválasztás

Feladat:

Adott tulajdonságú elemek maximuma – **feltételes maximumkeresés**.

Kiválogatás + maximum-kiválasztás

Specifikáció:

- Bemenet: $N \in \mathbb{N}$, $X \in H^N$
- Kimenet: $Van \in \mathbb{L}$, $Ind \in \mathbb{N}$
- Előfeltétel: –
- Utófeltétel: $Van = \exists i\ (1 \le i \le N): T(X_i)$ és $Van \rightarrow 1 \le Ind \le N$ és $T(X_{Ind})$

Feladat:

Adott tulajdonságú elemek maximuma – **feltételes maximumkeresés**.

Specifikáció:

- Bemenet: $N \in \mathbb{N}$,
  $X \in H^N$
- Kimenet: $MaxI \in \mathbb{N}$, $Van \in \mathbb{L}$
- Előfeltétel: –
- Utófeltétel: $Van = \exists i\ (1 \le i \le N): T(X_i)$ és
  $Van \rightarrow (\ 1 \le MaxI \le N$ és $T(X_{MaxI})$ és
  $\forall i (1 \le i \le N): T(X_i) \rightarrow X_{MaxI} \ge X_i\ )$

Specifikáció:

- Bemenet: $N \in \mathbb{N}$, $X \in H^N$
- Kimenet: $MaxI \in \mathbb{N}$, $Van \in \mathbb{L}$
- Előfeltétel: –
- Utófeltétel: $Van = \exists i\ (1 \le i \le N): T(X_i)$ és $Van \to (1 \le MaxI \le N$ és $T(X_{MaxI})$ és $\forall i (1 \le i \le N): T(X_i) \to X_{MaxI} \ge X_i)$

Kiválogatás + maximum-kiválasztás

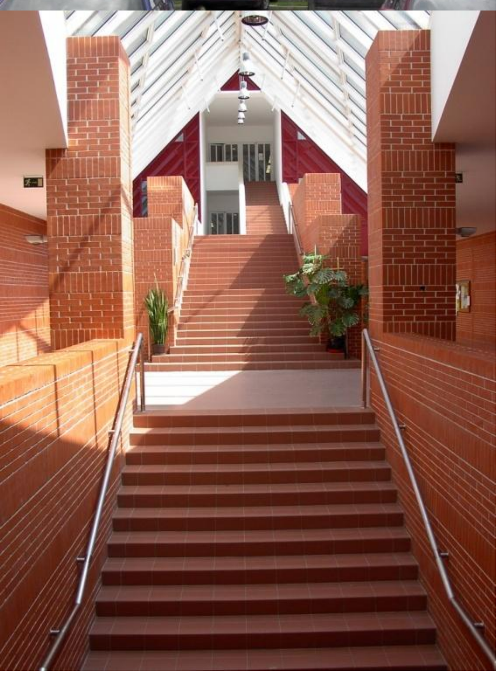
Specifikáció$_2$:

- Utófeltétel$_2$: $(Van, MaxI) = \underset{\substack{i=1 \\ T(X_i)}}{\overset{N}{\mathrm{MaxInd}}}\ X_i$

Specifikáció:

- Bemenet: $N \in \mathbb{N}$, $X \in H^N$
- Kimenet: $MaxI \in \mathbb{N}$, $Van \in \mathbb{L}$
- Előfeltétel: –
- Utófeltétel: $Van = \exists i\ (1 \le i \le N): T(X_i)$ és $Van \rightarrow (1 \le MaxI \le N$ és $T(X_{MaxI})$ és $\forall i (1 \le i \le N): T(X_i) \rightarrow X_{MaxI} \ge X_i)$

Kiválogatás + maximum-kiválasztás



Specifikáció$_2$:

- Utófeltétel$_2$: $(Van, MaxI) = \underset{\substack{i=1 \\ T(X_i)}}{\overset{N}{\mathrm{MaxInd}}}\, X_i$

Specifikáció$_3$:

- Utófeltétel$_3$: $(Van, Ind, Ért) = \underset{\substack{i=1 \\ T(X_i)}}{\overset{N}{\mathrm{Max}}}\, X_i$

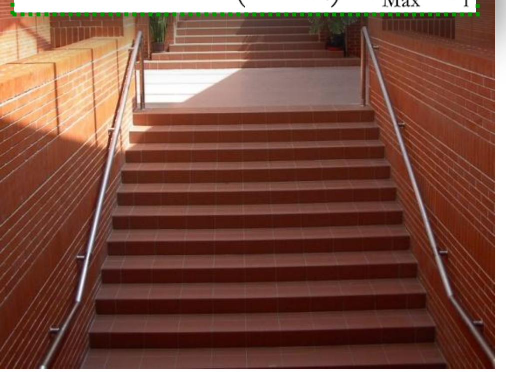
Kiválogatás + maximum-kiválasztás

Specifikáció:

- Bemenet: $N \in \mathbb{N}$
  $X \in H^N$
  $T: H \to L$
- Kimenet: $Db \in \mathbb{N}$
  $Y \in \mathbb{N}^{Db}$
- 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
  $Y \subseteq (1,2,\ldots,N)$

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
  $\forall i\ (1 \le i \le N): X_{Max} \ge X_i$



A megoldás felé:

Specifikáció':

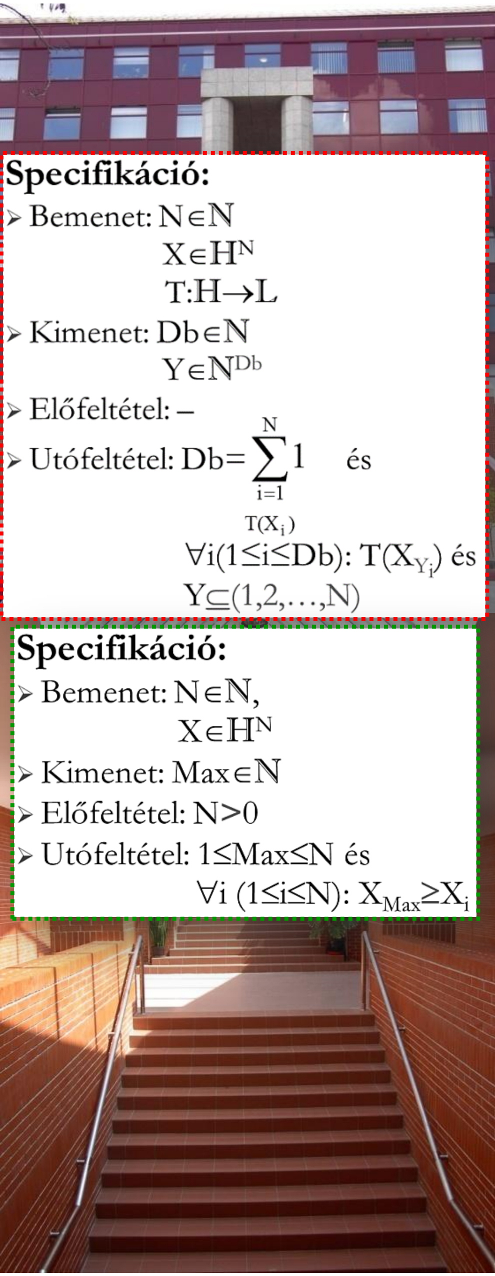
- Utófeltétel': $(Db,Y) = \underset{\substack{i=1 \\ T(X_i)}}{\overset{N}{\text{Kiválogat}}}\ i$ és

  $Van = Db>0$ és

  $Van \to (1 \le MaxI \le N$ és $T(X_{MaxI})$ és

  $MaxI = \underset{i=1}{\overset{Db}{\text{MaxInd}}}\ X_{Y_i}\ )$

Kiválogatás + maximum-kiválasztás



Specifikáció:

- Bemenet: $N \in \mathbb{N}$
  $X \in H^N$
  $T: H \to L$
- Kimenet: $Db \in \mathbb{N}$
  $Y \in \mathbb{N}^{Db}$
- 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
  $Y \subseteq (1,2,\ldots,N)$

Specifikáció:

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

A megoldás felé:

Specifikáció’:

- Utófeltétel’: $(Db, Y) = \underset{\substack{i=1 \\ T(X_i)}}{\overset{N}{\text{Kiválogat}}}\ i$ és

$$Van = Db > 0 \quad \text{és}$$

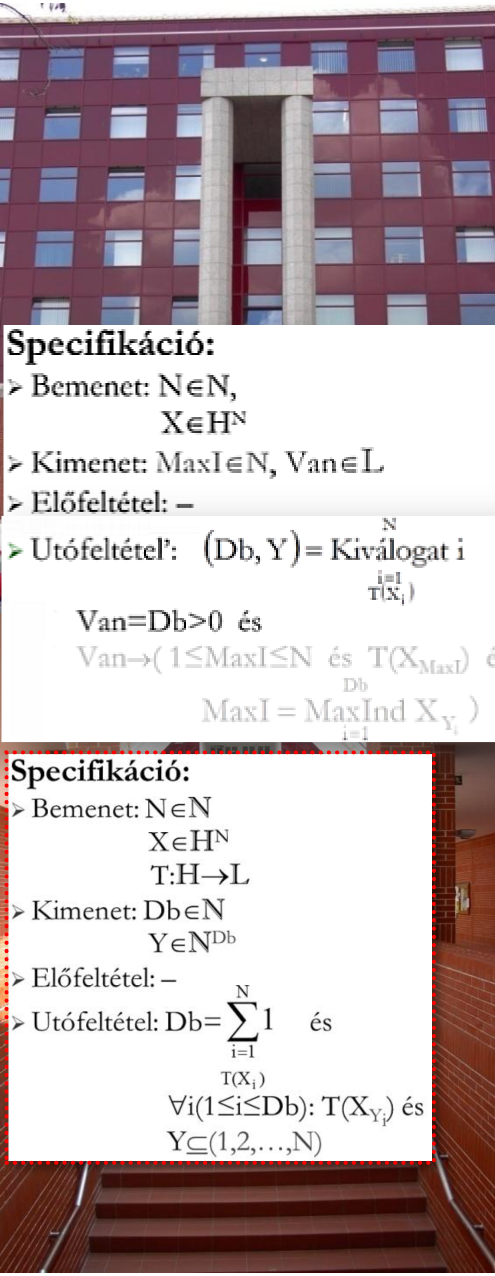
$$Van \to \left(1 \leq MaxI \leq N \quad \text{és} \quad T(X_{MaxI}) \quad \text{és} \right.$$

$$\left. MaxI = \underset{i=1}{\overset{Db}{\text{MaxInd}}}\ X_{Y_i} \right)$$

Kiolvasható az algoritmikus ötlet:

Válogassuk ki az adott tulajdonságúakat, majd keressünk maximumot, ha van értelme!

Kiválogatás + maximum-kiválasztás



Specifikáció:

- Bemenet: $N \in \mathbb{N}$, $X \in H^N$
- Kimenet: $MaxI \in \mathbb{N}$, $Van \in \mathbb{L}$
- Előfeltétel: –
- Utófeltétel': $(Db, Y) = \underset{\substack{i=1 \\ T(X_i)}}{\overset{N}{\text{Kiválogat}}}\, i$ és
  $Van = Db > 0$ és
  $Van \to (1 \le MaxI \le N$ és $T(X_{MaxI})$ és
  $MaxI = \underset{i=1}{\overset{Db}{\text{MaxInd}}}\, X_{Y_i})$

Specifikáció:

- Bemenet: $N \in \mathbb{N}$
  $X \in H^N$
  $T: H \to \mathbb{L}$
- Kimenet: $Db \in \mathbb{N}$
  $Y \in \mathbb{N}^{Db}$
- 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
  $Y \subseteq (1, 2, \ldots, N)$

1. megoldás algoritmusa:

Válogassuk ki az adott tulajdonságúakat, majd …!

Kiválogatás + maximum-kiválasztás

1. megoldás algoritmusa:

Válogassuk ki az adott tulajdonságúakat, majd …!

Specifikáció:

- Bemenet: $N \in \mathbb{N}$, $X \in H^N$
- Kimenet: $MaxI \in \mathbb{N}$, $Van \in \mathbb{L}$
- Előfeltétel: –
- Utófeltétel': $(Db, Y) = \underset{\substack{i=1 \\ T(X_i)}}{\overset{N}{\text{Kiválogat}}}\ i$ és

  $Van = Db > 0$ és

  $Van \rightarrow (1 \le MaxI \le N$ és $T(X_{MaxI})$ és

  $MaxI = \underset{i=1}{\overset{Db}{\text{MaxInd}}}\ X_{Y_i})$

Specifikáció:

- Bemenet: $N \in \mathbb{N}$
  $X \in H^N$
  $T: H \rightarrow \mathbb{L}$
- Kimenet: $Db \in \mathbb{N}$
  $Y \in \mathbb{N}^{Db}$
- 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	—

Kiválogatás + maximum-kiválasztás

Specifikáció:
- Bemenet: $N \in \mathbb{N}$, $X \in H^N$
- Kimenet: $MaxI \in \mathbb{N}$, $Van \in \mathbb{L}$
- Előfeltétel: –
- Utófeltétel': $(Db, Y) = \text{Kiválogat}_{\substack{i=1 \\ T(X_i)}}^{N}\ i$ és
  $Van = Db > 0$ és
  $Van \rightarrow (1 \le MaxI \le N$ és $T(X_{MaxI})$ és
  $MaxI = \text{MaxInd}_{i=1}^{Db}\ X_{Y_i})$

Specifikáció:
- Bemenet: $N \in \mathbb{N}$
  $X \in H^N$
  $T: H \rightarrow \mathbb{L}$
- Kimenet: $Db \in \mathbb{N}$
  $Y \in \mathbb{N}^{Db}$
- 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: Db:=Db+1; Y[Db]:=i — N: —

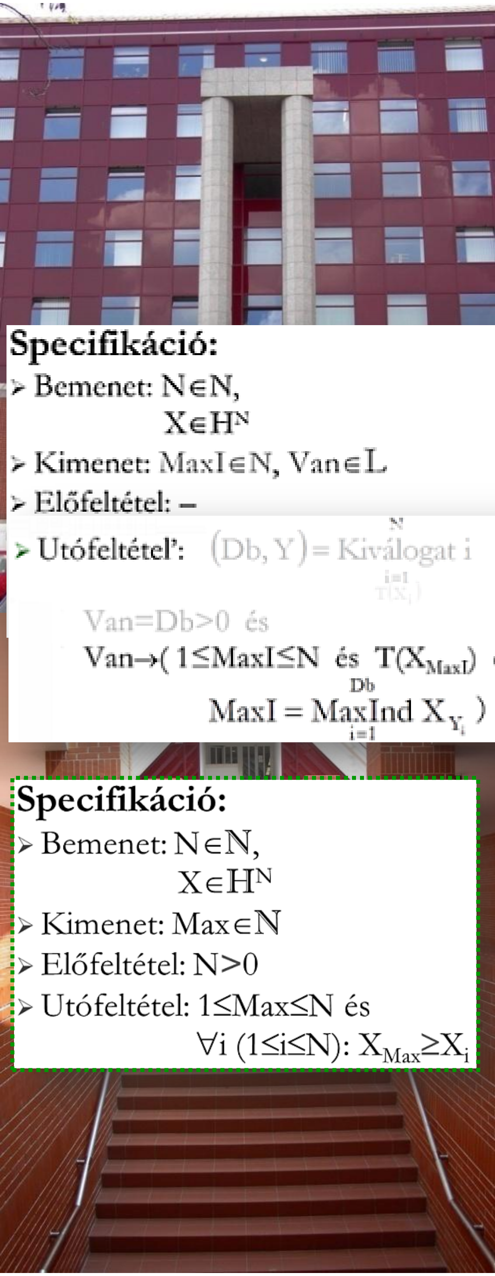
1. megoldás algoritmusa:

Válogassuk ki az adott tulajdonságúakat, majd …!

Változó
i,Db:**Egész**
Y:Tömb[…]

Db:=0
i=1..N
T(X[i]) — I: Db:=Db+1; Y[Db]:=i — N: —
Van:=Db>0
…

Kiválogatás + maximum-kiválasztás



Specifikáció:

- Bemenet: $N\in N$,
  $X\in H^N$
- Kimenet: $MaxI\in N$, $Van\in L$
- Előfeltétel: –
- Utófeltétel': $(Db,Y)=\underset{\substack{i=1\\T(X_i)}}{\overset{N}{\text{Kiválogat}}}\ i$ és
  $Van=Db>0$ és
  $Van\rightarrow(1\le MaxI\le N$ és $T(X_{MaxI})$ és
  $MaxI=\underset{i=1}{\overset{Db}{\text{MaxInd}}}\ X_{Y_i})$

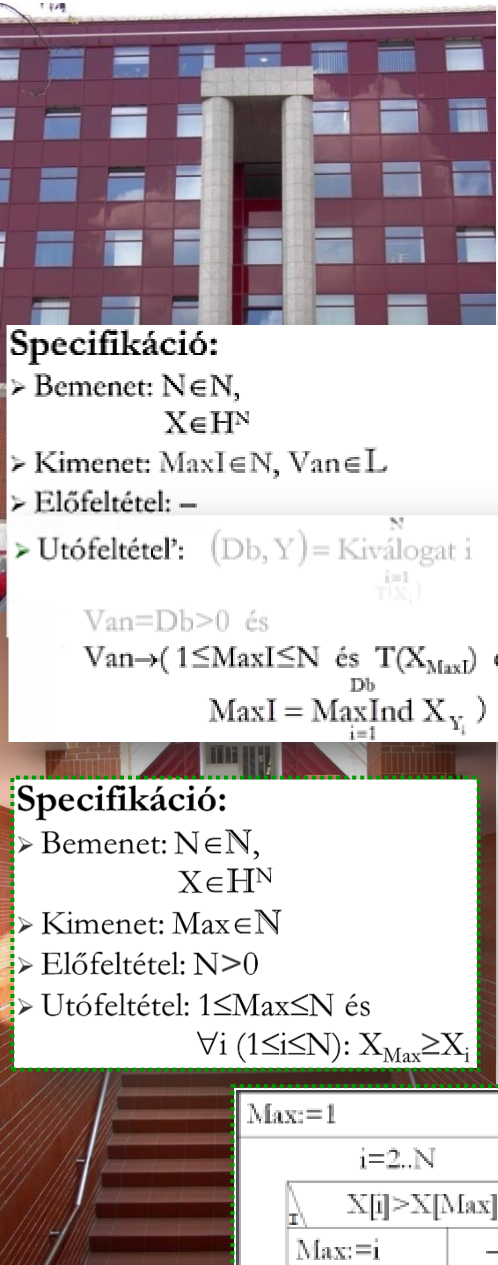
Specifikáció:

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

1. megoldása algoritmusa:

… , majd keressünk maximumot, ha van értelme!

Kiválogatás + maximum-kiválasztás



1. megoldása algoritmusa:

… , majd keressünk maximumot, ha van értelme!

Specifikáció:

- Bemenet: $N \in \mathbb{N}$,
  $X \in H^N$
- Kimenet: $MaxI \in \mathbb{N}$, $Van \in \mathbb{L}$
- Előfeltétel: –
- Utófeltétel': $(Db, Y) = \underset{\substack{i=1\\T(X_i)}}{\overset{N}{\text{Kiválogat}}}\ i$ és
  $Van = Db > 0$ és
  $Van \rightarrow (1 \le MaxI \le N$ és $T(X_{MaxI})$ és
  $MaxI = \underset{i=1}{\overset{Db}{\text{MaxInd}}}\ X_{Y_i}\ )$

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:=1	
i=2..N	
X[i]>X[Max] (I / N)	
Max:=i	—

Kiválogatás + maximum-kiválasztás

1. megoldása algoritmusa:

… , majd keressünk maximumot, ha van értelme!

Specifikáció:
- Bemenet: $N \in \mathbb{N}$, $X \in H^N$
- Kimenet: $MaxI \in \mathbb{N}$, $Van \in \mathbb{L}$
- Előfeltétel: –
- Utófeltétel': $(Db, Y) = \underset{\substack{i=1\\T(X_i)}}{\overset{N}{\text{Kiválogat}}}\; i$ és
  $Van = Db > 0$ és
  $Van \rightarrow (1 \le MaxI \le N$ és $T(X_{MaxI})$ és
  $MaxI = \underset{i=1}{\overset{Db}{\text{MaxInd}}}\; X_{Y_i})$

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:=1
i=2..N
  X[i]>X[Max]
  I: Max:=i    N: —
```

```
…
Van
I:
  MaxI:=Y[1]
  i=2..Db
    X[Y[i]]>X[MaxI]
    I: MaxI:=Y[i]    N: —
N:
  —
```

Kiválogatás + maximum-kiválasztás

Specifikáció:

- Bemenet: $N \in \mathbb{N}$, $X \in H^N$
- Kimenet: $MaxI \in \mathbb{N}$, $Van \in \mathbb{L}$
- Utófeltétel': $(Db, Y) = \underset{\substack{i=1 \\ T(X_i)}}{\overset{N}{\text{Kiválogat}}}\ i$ és $Van = Db > 0$ és $Van \rightarrow (1 \le MaxI \le N$ és $T(X_{MaxI})$ és $MaxI = \underset{i=1}{\overset{Db}{\text{MaxInd}}}\ X_{Y_i})$

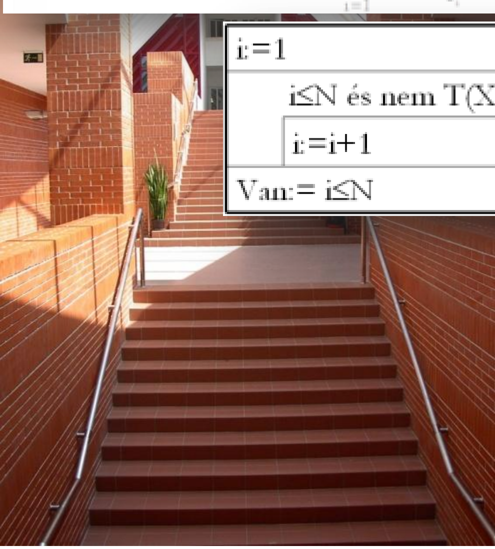
2. megoldási ötlet (és algoritmusa)**:**

Induljunk ki a specifikációban észrevett téte-lekből: a kiválogatás helyett keressük meg az első T-tulajdonságút, …

Kiválogatás + maximum-kiválasztás

Specifikáció:
- Bemenet: $N \in \mathbb{N}$, $X \in H^N$
- Kimenet: $MaxI \in \mathbb{N}$, $Van \in \mathbb{L}$
- Utófeltétel': $(Db, Y) = \underset{\substack{i=1 \\ T(X_i)}}{\overset{N}{\text{Kiválogat}}}\ i$ és
  $Van = Db > 0$ és
  $Van \rightarrow (1 \le MaxI \le N$ és $T(X_{MaxI})$ és
  $MaxI = \underset{i=1}{\overset{Db}{\text{MaxInd}}}\ X_{Y_i}$)

i:=1
i≤N és nem T(X[i])
i:=i+1
Van:= i≤N



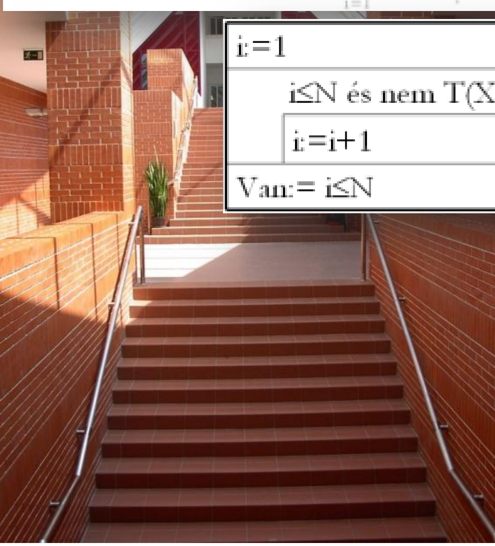
2. megoldási ötlet (és algoritmusa)**:**

Induljunk ki a specifikációban észrevett tételekből: a kiválogatás helyett keressük meg az első T-tulajdonságút, …

Kiválogatás + maximum-kiválasztás

Specifikáció:
- Bemenet: $N \in \mathbb{N}$, $X \in H^N$
- Kimenet: $MaxI \in \mathbb{N}$, $Van \in \mathbb{L}$
- Utófeltétel': $(Db, Y) = \underset{\substack{i=1 \\ T(X_i)}}{\overset{N}{\text{Kiválogat}}}\ i$ és
  $Van = Db > 0$ és
  $Van \rightarrow (1 \le MaxI \le N$ és $T(X_{MaxI})$ és $MaxI = \underset{i=1}{\overset{Db}{\text{MaxInd}}}\ X_{Y_i})$

i:=1
i≤N és nem T(X[i])
i:=i+1
Van:= i≤N



2. megoldási ötlet (és algoritmusa):

Induljunk ki a specifikációban észrevett tételekből: a kiválogatás helyett keressük meg az első T-tulajdonságút, …

i:=1
i≤N és nem T(X[i])
i:=i+1
Van:=i≤N
…

Változó
i:**Egész**

Kiválogatás + maximum-kiválasztás

Specifikáció:

- Bemenet: $N \in \mathbb{N}$, $X \in H^N$
- Kimenet: $MaxI \in \mathbb{N}$, $Van \in \mathbb{L}$
- Utófeltétel': $(Db, Y) = \underset{\substack{i=1 \\ T(X_i)}}{\overset{N}{\text{Kiválogat}}}\ i$ és

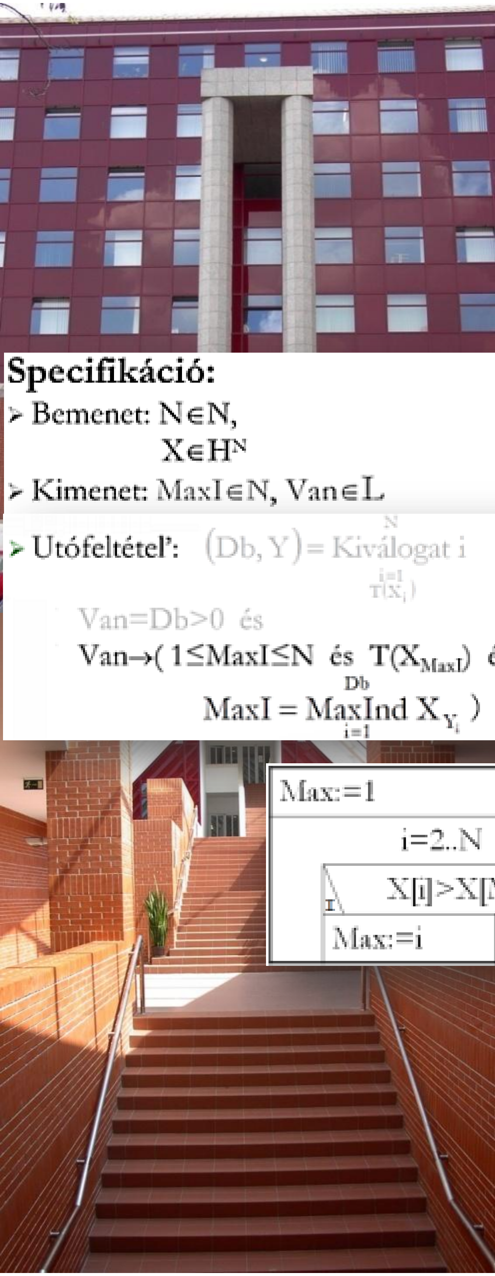
  $Van = Db > 0$ és

  $Van \rightarrow (1 \leq MaxI \leq N$ és $T(X_{MaxI})$ és $MaxI = \underset{i=1}{\overset{Db}{\text{MaxInd}}}\ X_{Y_i})$

2. megoldási ötlet (és algoritmusa)**:**

… majd válasszuk ki az **ilyenek** maximumát!

Kiválogatás + maximum-kiválasztás



Specifikáció:

- Bemenet: $N \in \mathbb{N}$, $X \in H^N$
- Kimenet: $MaxI \in \mathbb{N}$, $Van \in \mathbb{L}$
- Utófeltétel': $(Db, Y) = \underset{\substack{i=1 \\ T(X_i)}}{\overset{N}{\text{Kiválogat}}}\ i$ és

  $Van = Db > 0$ és

  $Van \to (1 \leq MaxI \leq N$ és $T(X_{MaxI})$ és

  $MaxI = \underset{i=1}{\overset{Db}{\text{MaxInd}}}\ X_{Y_i})$

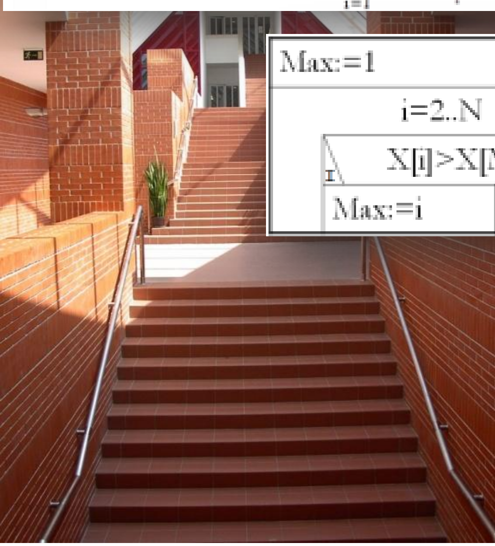
Max:=1	
i=2..N	
I — X[i]>X[Max] — N	
Max:=i	—

2. megoldási ötlet (és algoritmusa)**:**

… majd válasszuk ki az **ilyenek** maximumát!

Kiválogatás + maximum-kiválasztás

Specifikáció:
- Bemenet: $N \in \mathbb{N}$, $X \in H^N$
- Kimenet: $MaxI \in \mathbb{N}$, $Van \in \mathbb{L}$
- Utófeltétel': $(Db, Y) = \text{Kiválogat}_{\substack{i=1 \\ T(X_i)}}^{N}\ i$ és
  $Van = Db > 0$ és
  $Van \rightarrow (1 \le MaxI \le N$ és $T(X_{MaxI})$ és
  $MaxI = \text{MaxInd}_{i=1}^{Db}\ X_{Y_i})$



Max:=1	
i=2..N	
X[i]>X[Max] (I / N)	
Max:=i	—

2. megoldási ötlet (és algoritmusa):

… majd válasszuk ki az **ilyenek** maximumát!

…

Van (I / N)

- I:
  - MaxI:=i
  - i=i+1..N
    - T(X[i]) (I / N)
      - I:
        - X[i]>X[MaxI] (I / N)
          - I: MaxI:=i
          - N: —
      - N: —
- N: —

„ilyenek”

Kiválogatás + maximum-kiválasztás

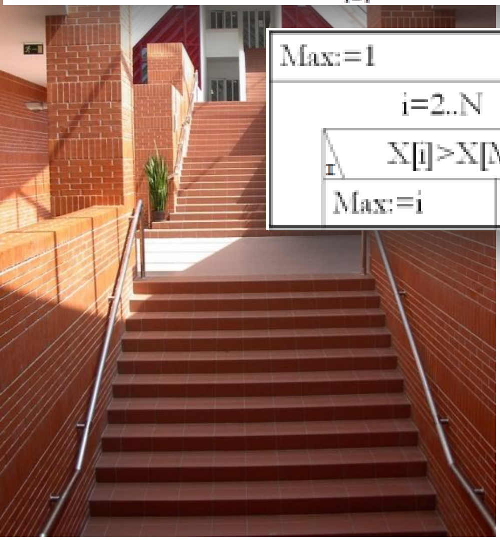
2. megoldási ötlet (és algoritmusa):

… majd válasszuk ki az **ilyenek** maximumát!

Specifikáció:
- Bemenet: $N \in \mathbb{N}$, $X \in H^N$
- Kimenet: $MaxI \in \mathbb{N}$, $Van \in \mathbb{L}$
- Utófeltétel': $(Db, Y) = \underset{\substack{i=1 \\ T(X_i)}}{\overset{N}{\text{Kiválogat}}}\ i$ és

  $Van = Db > 0$ és

  $Van \rightarrow (1 \leq MaxI \leq N$ és $T(X_{MaxI})$ és $MaxI = \underset{i=1}{\overset{Db}{\text{MaxInd}}}\ X_{Y_i})$



Max:=1
i=2..N
X[i]>X[Max] — I: Max:=i | N: —

…

Van
- I:
  - MaxI:=i
  - i=i+1..N
    - T(X[i])
      - I: X[i]>X[MaxI]
        - I: MaxI:=i
        - N: —
      - N: —
- N: —

Kiválogatás + maximum-kiválasztás

2. megoldási ötlet (és algoritmusa):

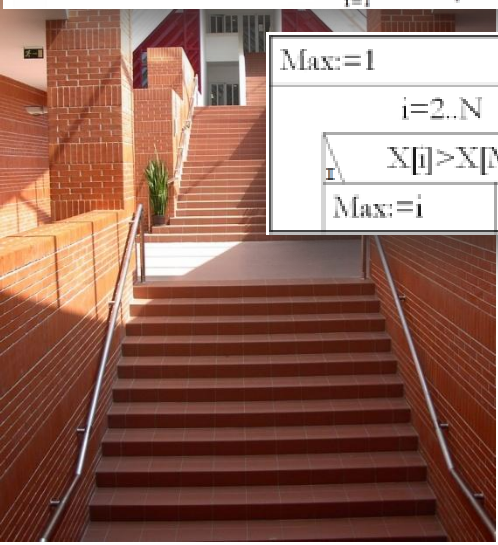
… majd válasszuk ki az **ilyenek** maximumát!

Specifikáció:

- Bemenet: $N \in \mathbb{N}$, $X \in H^N$
- Kimenet: $MaxI \in \mathbb{N}$, $Van \in \mathbb{L}$
- Utófeltétel': $(Db, Y) = \underset{\substack{i=1 \\ T(X_i)}}{\overset{N}{\text{Kiválogat}}}\ i$ és

  $Van = Db > 0$ és

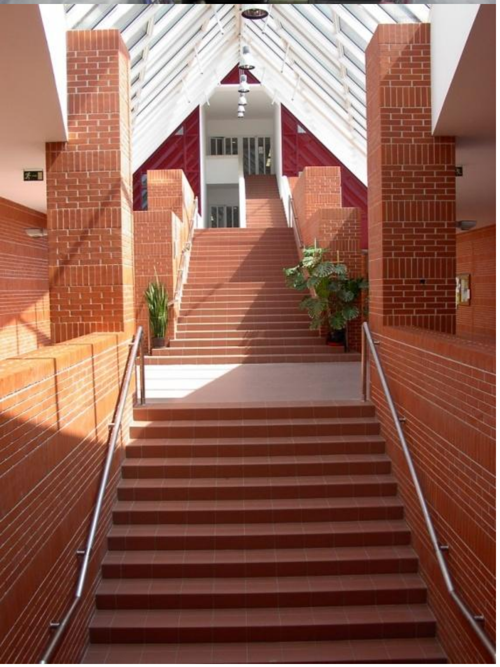
  $Van \to (1 \le MaxI \le N$ és $T(X_{MaxI})$ és $MaxI = \underset{i=1}{\overset{Db}{\text{MaxInd}}}\ X_{Y_i})$



```
Max:=1
i=2..N
    X[i]>X[Max]
    I: Max:=i  |  N: —
```

```
...
---------------------------------------------
Van
I:
  MaxI:=i
  i=i+1..N
      T(X[i]) és X[i]>X[MaxI]
      I: MaxI:=i  |  N: —
N:
  —
```

Kiválogatás + maximum-kiválasztás



3. megoldási ötlet (és algoritmusa)**:**

Kiválogatás, ill. keresés helyett azonnal válasszuk ki a maximumot!

Kell egy fiktív **0. elem** a maximum-kiválasztáshoz, ami **kisebb minden** „normál” elemnél.

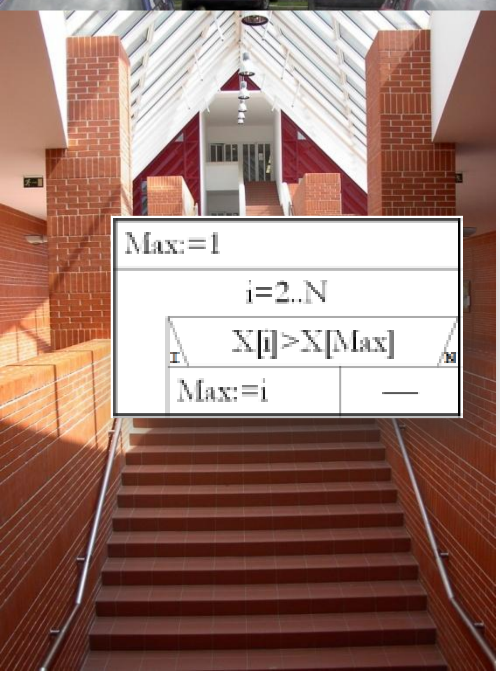
Kiválogatás + maximum-kiválasztás

3. megoldási ötlet (és algoritmusa)**:**

Kiválogatás, ill. keresés helyett azonnal válasszuk ki a maximumot!

Kell egy fiktív **0. elem** a maximum-kiválasztáshoz, ami **kisebb minden** „normál" elemnél.

Kiválogatás + maximum-kiválasztás



3. megoldási ötlet (és algoritmusa)**:**

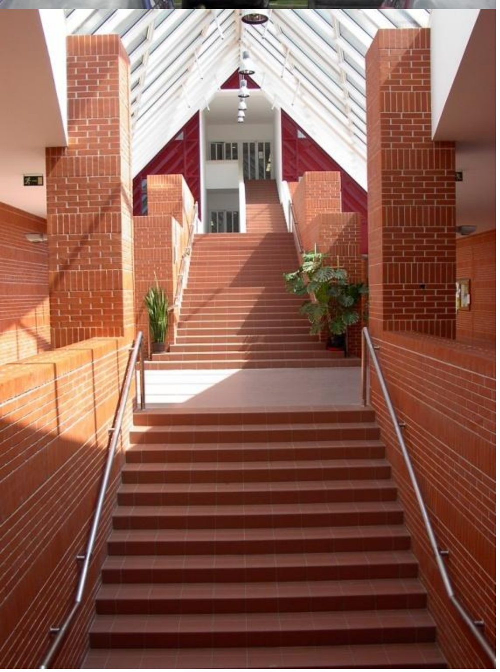
Kiválogatás, ill. keresés helyett azonnal válasszuk ki a maximumot!

Kell egy fiktív **0. elem** a maximum-kiválasztáshoz, ami **kisebb minden** „normál” elemnél.

Változó
i:**Egész**

X[0]:=-∞		
MaxI:=0		
i=1..N		
	I — T(X[i]) és X[i]>X[MaxI] — N	
	MaxI:=i	—
Van:=MaxI>0		

Maximum-kiválasztás + kiválogatás



Feladat:

Összes maximális elem kiválogatása.

Maximum-kiválasztás + kiválogatás

Specifikáció:

- Bemenet: $N \in \mathbb{N}$
  $X \in H^N$
  $T: H \to L$
- Kimenet: $Db \in \mathbb{N}$
  $Y \in \mathbb{N}^{Db}$
- 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
  $Y \subseteq (1,2,\ldots,N)$

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
  $\forall i\ (1 \le i \le N): X_{Max} \ge X_i$

Feladat:

Összes maximális elem kiválogatása.

Specifikáció:

- Bemenet: $N \in \mathbb{N},\ X \in H^N$
- Kimenet: $Db \in \mathbb{N}$,
  $\boxed{MaxI} \in \mathbb{N}^{Db}$
- Előfeltétel: $\boxed{N>0}$
- Utófeltétel: $Db = \sum_{\substack{i=1 \\ X_i = X_{MaxI_1}}}^{N} 1$ és

$$\forall i(1 \le i \le Db): \boxed{\forall j(1 \le j \le N): X_{MaxI_i} \ge X_j \text{ és}}$$

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

Maximum-kiválasztás + kiválogatás

Specifikáció:
- Bemenet: $N \in \mathbb{N}$
  $X \in H^N$
  $T: H \rightarrow L$
- Kimenet: $Db \in \mathbb{N}$
  $Y \in \mathbb{N}^{Db}$
- 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
  $Y \subseteq (1,2,\ldots,N)$

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
  $\forall i\ (1 \le i \le N): X_{Max} \ge X_i$

Feladat:

Összes maximális elem kiválogatása.

Specifikáció:

- Bemenet: $N \in \mathbb{N}, X \in H^N$
- Kimenet: $Db \in \mathbb{N}$,
  $MaxI \in \mathbb{N}^{Db}$
- Előfeltétel: $N>0$
- Utófeltétel: $MaxÉ = \underset{i=1}{\overset{N}{\text{MaxÉrt}}}\ X_i$ és

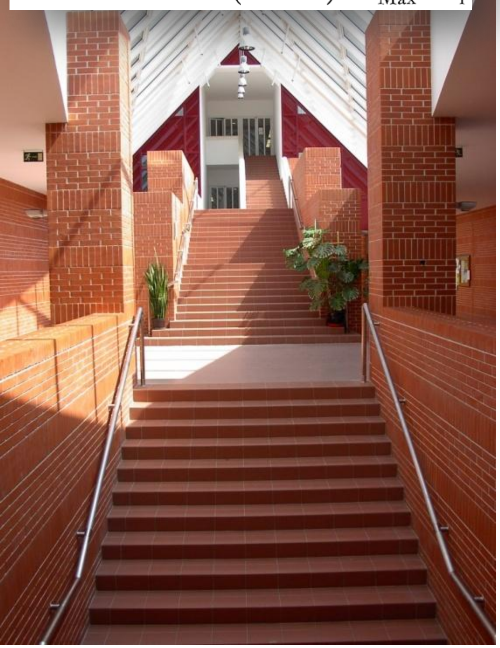
$$(Db, MaxI) = \underset{\substack{i=1 \\ X_i = MaxÉ}}{\overset{N}{\text{Kiválogat}}}\ i$$

Specifikáció:

- Bemenet: $N \in \mathbb{N}, X \in H^N$
- Kimenet: $Db \in \mathbb{N}$, $MaxI \in \mathbb{N}^{Db}$
- Előfeltétel: $N>0$
- Utófeltétel: $MaxÉ = \underset{i=1}{\overset{N}{MaxÉrt}}\, X_i$ és $(Db, MaxI) = \underset{\substack{i=1 \\ X_i=MaxÉ}}{\overset{N}{Kiválogat}}\, i$

Specifikáció:

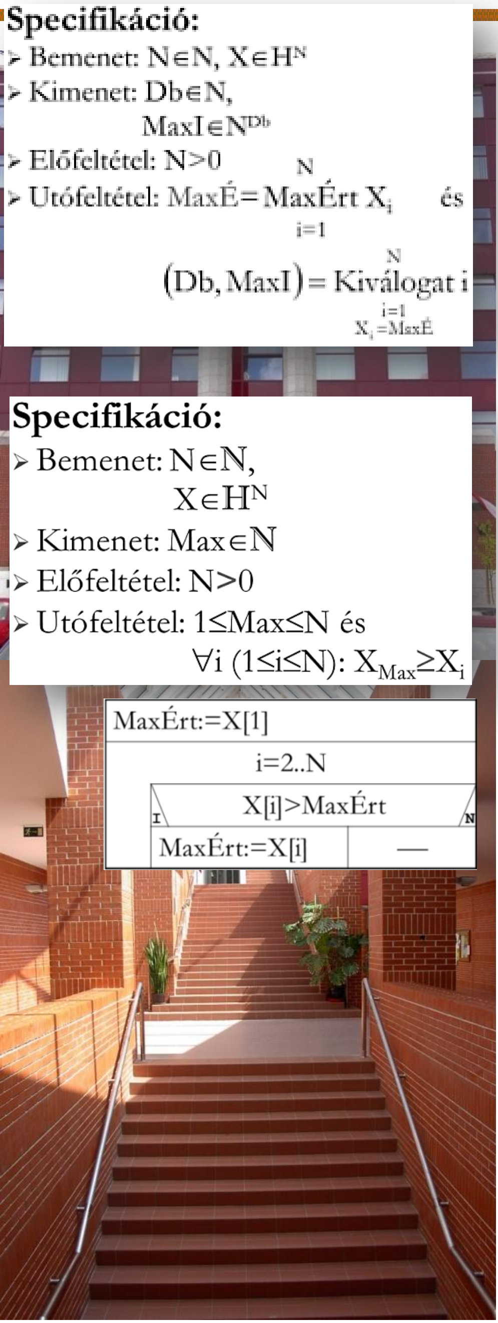
- Bemenet: $N \in \mathbb{N}$, $X \in H^N$
- Kimenet: $Max \in \mathbb{N}$
- Előfeltétel: $N>0$
- Utófeltétel: $1 \le Max \le N$ és $\forall i\ (1 \le i \le N): X_{Max} \ge X_i$



Maximum-kiválasztás + kiválogatás

1. megoldási ötlet:

Határozzuk meg a maximumot, majd válogassuk ki a vele egyenlőeket!



Specifikáció:
- Bemenet: $N \in \mathbb{N}$, $X \in H^N$
- Kimenet: $Db \in \mathbb{N}$, $MaxI \in \mathbb{N}^{Db}$
- Előfeltétel: $N>0$
- Utófeltétel: $MaxÉ = \underset{i=1}{\overset{N}{MaxÉrt}}\, X_i$ és $(Db, MaxI) = \underset{\substack{i=1 \\ X_i = MaxÉ}}{\overset{N}{Kiválogat}}\, i$

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 $\forall i\ (1 \le i \le N)$: $X_{Max} \ge X_i$

MaxÉrt:=X[1]	
i=2..N	
X[i]>MaxÉrt (I / N)	
MaxÉrt:=X[i]	—

Maximum-kiválasztás + kiválogatás

1. megoldási ötlet:

Határozzuk meg a maximumot, majd válogassuk ki a vele egyenlőeket!

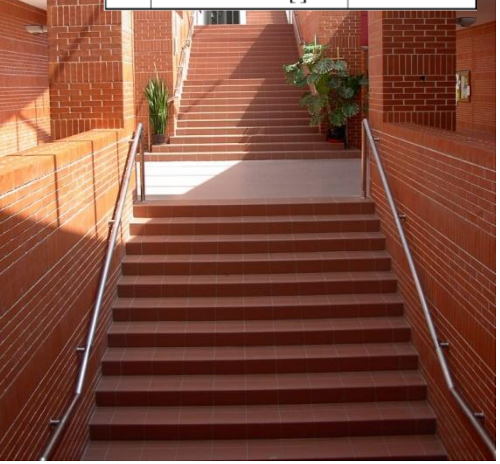
Specifikáció:

- Bemenet: $N \in \mathbb{N}$, $X \in H^N$
- Kimenet: $Db \in \mathbb{N}$, $MaxI \in \mathbb{N}^{Db}$
- Előfeltétel: $N>0$
- Utófeltétel: $MaxÉ = \underset{i=1}{\overset{N}{MaxÉrt}}\ X_i$ és $(Db, MaxI) = \underset{\substack{i=1 \\ X_i = MaxÉ}}{\overset{N}{Kiválogat}}\ i$

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 $\forall i\ (1 \le i \le N)$: $X_{Max} \ge X_i$

MaxÉrt:=X[1]	
i=2..N	
I X[i]>MaxÉrt N	
MaxÉrt:=X[i]	—



Maximum-kiválasztás + kiválogatás

1. megoldási ötlet:

Határozzuk meg a maximumot, majd válogassuk ki a vele egyenlőeket!

MaxÉ:=X[1]	
i=2..N	
I X[i]>MaxÉ N	
MaxÉ:=X[i]	—
...	

Változó

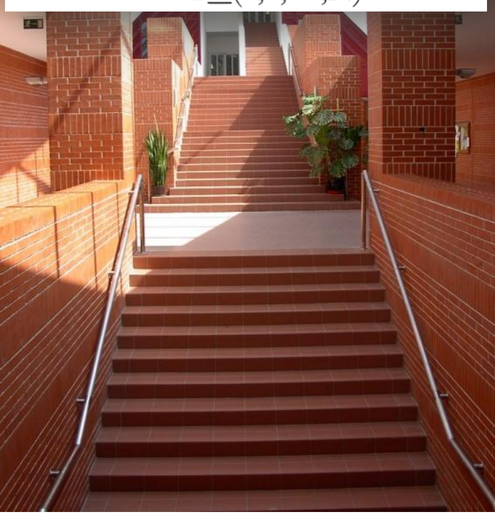
MaxÉ:TH

i:**Egész**

Maximum-kiválasztás + kiválogatás

Specifikáció:
- Bemenet: $N \in \mathbb{N}$, $X \in H^N$
- Kimenet: $Db \in \mathbb{N}$, $MaxI \in \mathbb{N}^{Db}$
- Előfeltétel: $N>0$
- Utófeltétel: $MaxÉ = \underset{i=1}{\overset{N}{\mathrm{MaxÉrt}}}\ X_i$ és $(Db, MaxI) = \underset{\substack{i=1 \\ X_i = MaxÉ}}{\overset{N}{\mathrm{Kiválogat}}}\ i$

Specifikáció:
- Bemenet: $N \in \mathbb{N}$, $X \in H^N$, $T: H \to L$
- Kimenet: $Db \in \mathbb{N}$, $Y \in \mathbb{N}^{Db}$
- 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 $Y \subseteq (1,2,\ldots,N)$



1. megoldási ötlet:

Határozzuk meg a maximumot, majd válogassuk ki a vele egyenlőeket!

Maximum-kiválasztás + kiválogatás

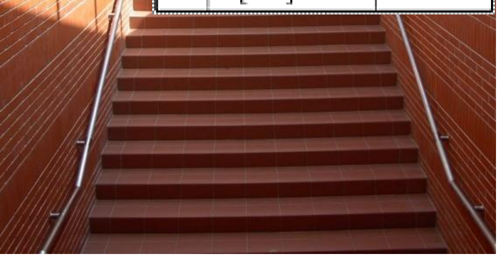
Specifikáció:

- Bemenet: $N \in \mathbb{N}$, $X \in H^N$
- Kimenet: $Db \in \mathbb{N}$, $MaxI \in \mathbb{N}^{Db}$
- Előfeltétel: $N>0$
- Utófeltétel: $MaxÉ = \underset{i=1}{\overset{N}{\mathrm{MaxÉrt}}}\, X_i$ és $(Db, MaxI) = \underset{\substack{i=1 \\ X_i = MaxÉ}}{\overset{N}{\mathrm{Kiválogat}}}\, i$

Specifikáció:

- Bemenet: $N \in \mathbb{N}$, $X \in H^N$, $T: H \to L$
- Kimenet: $Db \in \mathbb{N}$, $Y \in \mathbb{N}^{Db}$
- 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: Db:=Db+1
      Y[Db]:=i
   N: —
```



1. megoldási ötlet:

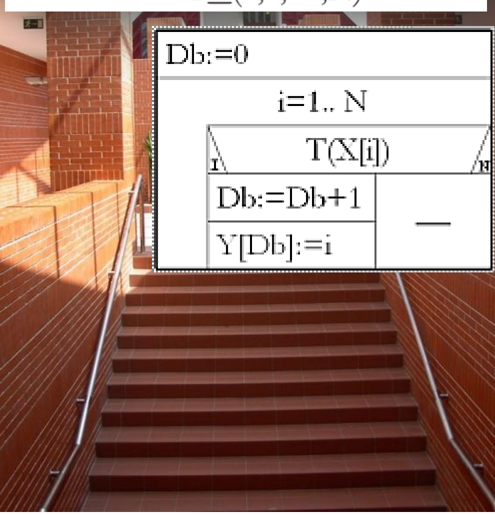
Határozzuk meg a maximumot, majd válogassuk ki a vele egyenlőeket!

Maximum-kiválasztás + kiválogatás

Specifikáció:
- Bemenet: $N \in \mathbb{N}$, $X \in H^N$
- Kimenet: $Db \in \mathbb{N}$, $MaxI \in \mathbb{N}^{Db}$
- Előfeltétel: $N>0$
- Utófeltétel: $MaxÉ = \underset{i=1}{\overset{N}{MaxÉrt}}\ X_i$ és $(Db, MaxI) = \underset{\substack{i=1 \\ X_i=MaxÉ}}{\overset{N}{Kiválogat}}\ i$

Specifikáció:
- Bemenet: $N \in \mathbb{N}$, $X \in H^N$, $T: H \to L$
- Kimenet: $Db \in \mathbb{N}$, $Y \in \mathbb{N}^{Db}$
- 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	



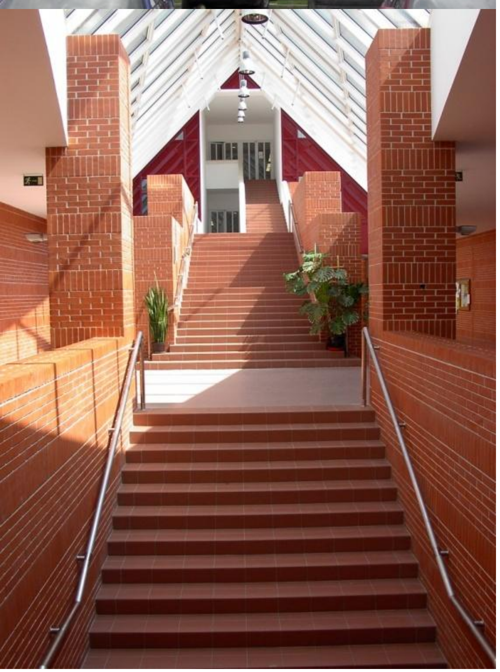
1. megoldási ötlet:

Határozzuk meg a maximumot, majd válogassuk ki a vele egyenlőeket!

…		
Db:=0		
i=1..N		
	X[i]=MaxÉ (I / N)	
	Db:=Db+1	—
	MaxI[Db]:=i	

Specifikáció:

- Bemenet: $N \in \mathbb{N}$, $X \in H^N$
- Kimenet: $Db \in \mathbb{N}$, $MaxI \in \mathbb{N}^{Db}$
- Előfeltétel: $N>0$
- Utófeltétel: $MaxÉ = \underset{i=1}{\overset{N}{MaxÉrt}}\ X_i$ és $(Db, MaxI) = \underset{\substack{i=1 \\ X_i=MaxÉ}}{\overset{N}{Kiválogat}}\ i$

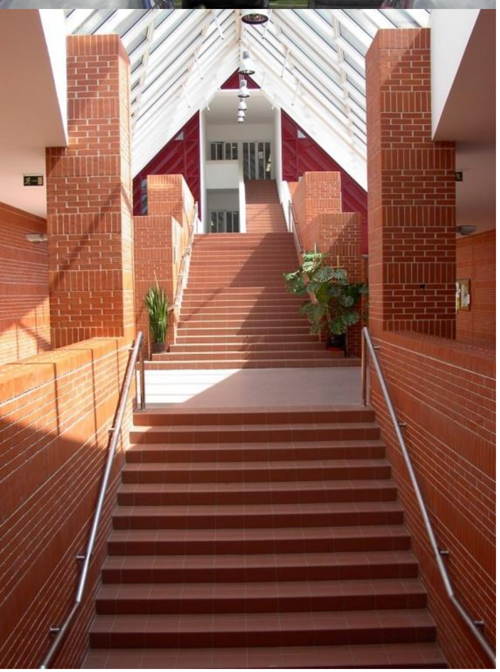


Maximum-kiválasztás + kiválogatás

2. megoldási ötlet:

A pillanatnyi maximálissal egyenlőeket azonnal válogassuk ki!

Specifikáció:
- Bemenet: $N \in \mathbb{N}, X \in H^N$
- Kimenet: $Db \in \mathbb{N}$, $MaxI \in \mathbb{N}^{Db}$
- Előfeltétel: $N>0$
- Utófeltétel: $MaxÉ = \underset{i=1}{\overset{N}{\text{MaxÉrt}}}\, X_i$ és $(Db, MaxI) = \underset{\substack{i=1 \\ X_i=MaxÉ}}{\overset{N}{\text{Kiválogat}}}\, i$



Maximum-kiválasztás + kiválogatás

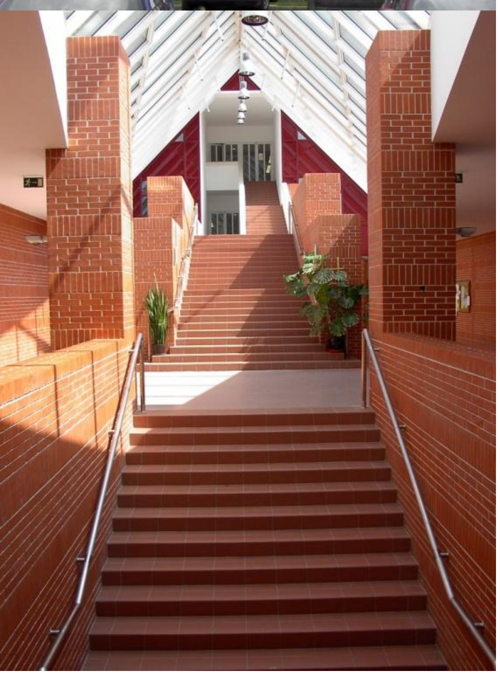
2. megoldási ötlet:

A pillanatnyi maximálissal egyenlőeket azonnal válogassuk ki!

Db:=1		
MaxI[1]:=1		
MaxÉ:=X[1]		
i=2..N		
	I X[i]>MaxÉ	I X[i]=MaxÉ
	Db:=1	Db:=Db+1
	MaxI[1]:=i	MaxI[Db]:=i
	MaxÉ:=X[i]	—

Változó
MaxÉ:TH
i:**Egész**

Eldöntés + megszámolás



Feladat:

Van-e egy sorozatban legalább K darab adott tulajdonságú elem?

Eldöntés + megszámolás

Feladat:

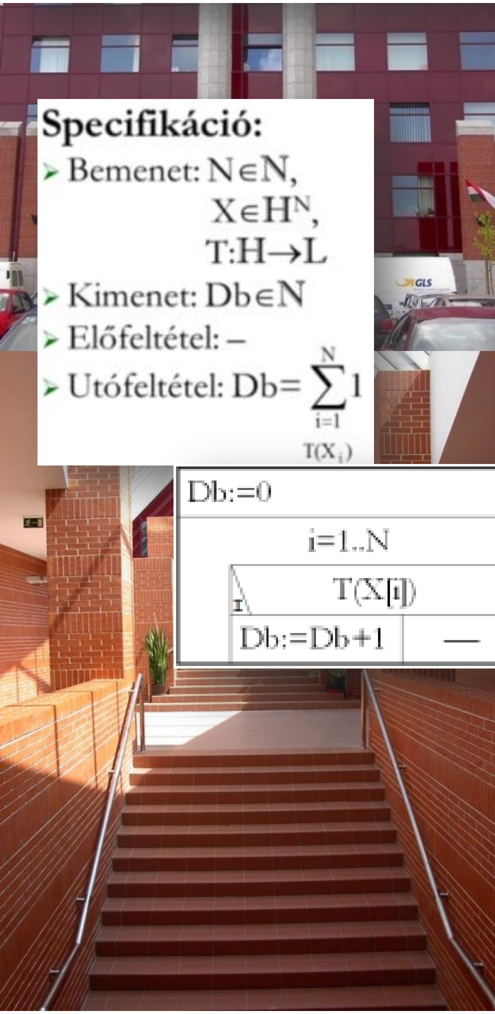
Van-e egy sorozatban legalább K darab adott tulajdonságú elem?

Specifikáció:

- Bemenet: $N,K \in \mathbb{N}$, $X \in H^N$
- Kimenet: $Van \in \mathbb{L}$
- Előfeltétel: – [$K>0$]
- Utófeltétel: $db = \sum_{\substack{i=1 \\ T(X_i)}}^{N} 1$ és $Van = db \geq K$

Specifikáció:

- Bemenet: $N,K \in \mathbb{N}$, $X \in H^N$
- Kimenet: $Van \in \mathbb{L}$
- Előfeltétel: – $[K>0]$
- Utófeltétel: $db = \sum_{\substack{i=1 \\ T(X_i)}}^{N} 1$ és $Van = db \geq K$



Specifikáció:

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

Db:=0	
i=1..N	
I T(X[i]) N	
Db:=Db+1	—

Eldöntés + megszámolás

1. megoldási ötlet:

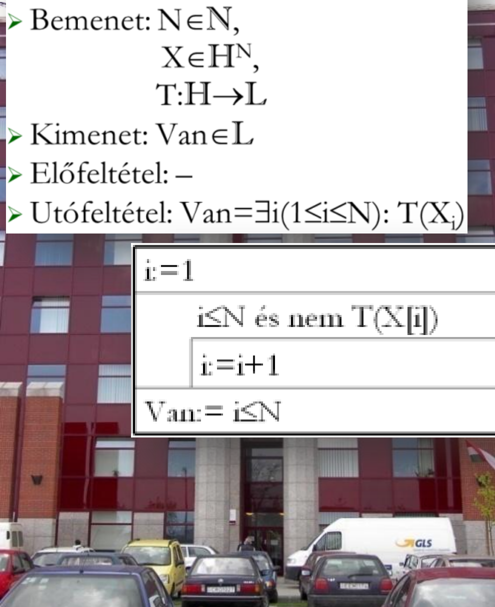
Számoljuk meg, hogy hány adott tulajdonságú van, majd nézzük meg, hogy ez legalább K-e! (Azaz valójában nincs: eldöntés tétel!)

db:=0	
i=1..N	
I T(X[i]) N	
db:=db+1	—
Van:=db≥K	

Változó
i:**Egész**

Specifikáció:

- Bemenet: $N \in \mathbb{N}$,
  $X \in H^N$,
  $T: H \to L$
- Kimenet: $Van \in L$
- Előfeltétel: –
- Utófeltétel: $Van = \exists i(1 \le i \le N): T(X_i)$

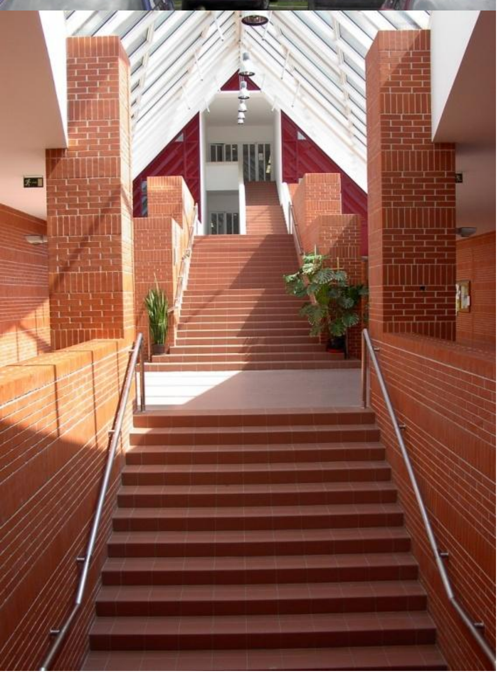


i:=1
i≤N és nem T(X[i])
i:=i+1
Van:= i≤N

Eldöntés + megszámolás

2. megoldási ötlet:

Ha már találtunk K darab adott tulajdonsá-gút, akkor ne nézzük tovább!



Specifikáció:

- Bemenet: $N \in \mathbb{N}$, $X \in H^N$, $T: H \to L$
- Kimenet: $Van \in L$
- Előfeltétel: –
- Utófeltétel: $Van = \exists i (1 \le i \le N): T(X_i)$

i:=1
i≤N és nem T(X[i])
i:=i+1
Van:= i≤N

i=1..N	
T(X[i])	
db:=db+1	—

Eldöntés + megszámolás

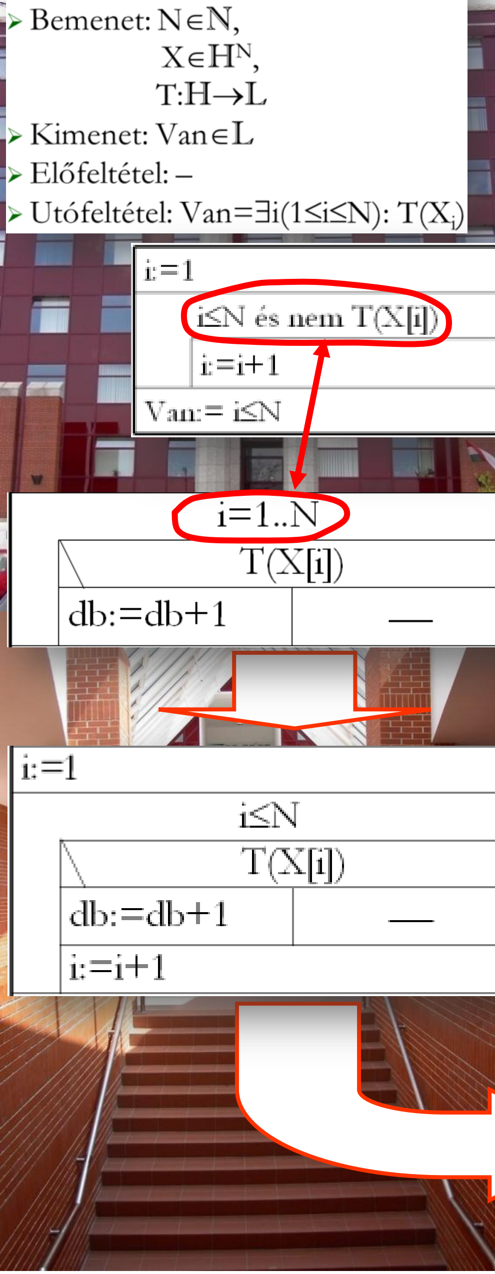
2. megoldási ötlet:

Ha már találtunk K darab adott tulajdonságút, akkor ne nézzük tovább!

Eldöntés + megszámolás

Specifikáció:

- Bemenet: $N \in \mathbb{N}$, $X \in H^N$, $T: H \to L$
- Kimenet: $Van \in L$
- Előfeltétel: –
- Utófeltétel: $Van = \exists i (1 \le i \le N): T(X_i)$



```
i:=1
  i≤N és nem T(X[i])
    i:=i+1
Van:= i≤N
```

```
i=1..N
  T(X[i])
    db:=db+1  |  —
```

```
i:=1
  i≤N
    T(X[i])
      db:=db+1  |  —
    i:=i+1
```

2. megoldási ötlet:

Ha már találtunk K darab adott tulajdonságút, akkor ne nézzük tovább!

Eldöntés + megszámolás

Specifikáció:
- Bemenet: $N \in \mathbb{N}$, $X \in H^N$, $T: H \to L$
- Kimenet: $Van \in L$
- Előfeltétel: –
- Utófeltétel: $Van = \exists i (1 \le i \le N): T(X_i)$

i:=1
i≤N és nem T(X[i])
i:=i+1
Van:= i≤N

i=1..N	
T(X[i])	
db:=db+1	—

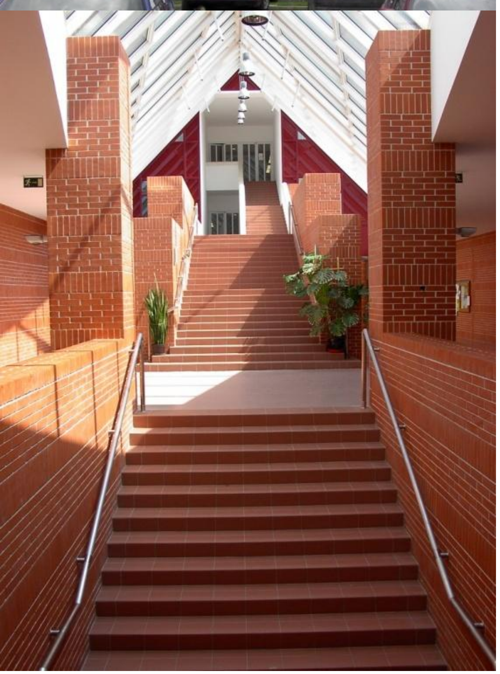
i:=1	
i≤N	
T(X[i])	
db:=db+1	—
i:=i+1	

2. megoldási ötlet:

Ha már találtunk K darab adott tulajdonságút, akkor ne nézzük tovább!

Változó
i:Egész

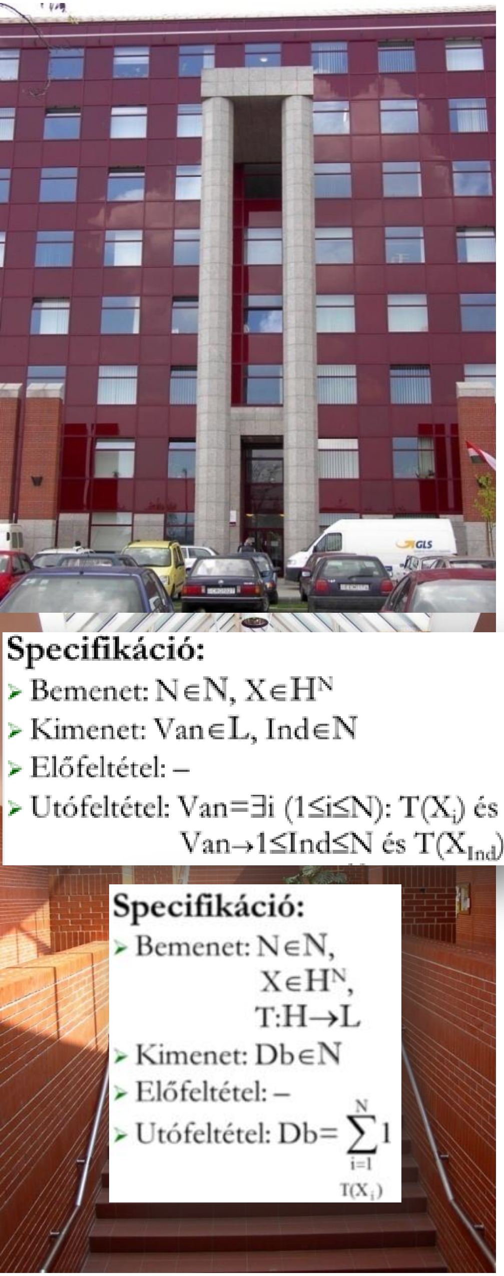
db:=0	
i:=1	
i≤N és db<K	
I — T(X[i]) — N	
db:=db+1	—
i:=i+1	
Van:=db=K	



Keresés + megszámolás

Feladat:

Egy sorozatban melyik a K. adott tulajdon-ságú elem (ha van egyáltalán)?



Keresés + megszámolás

Feladat:

Egy sorozatban melyik a K. adott tulajdonságú elem (ha van egyáltalán)?

Specifikáció:

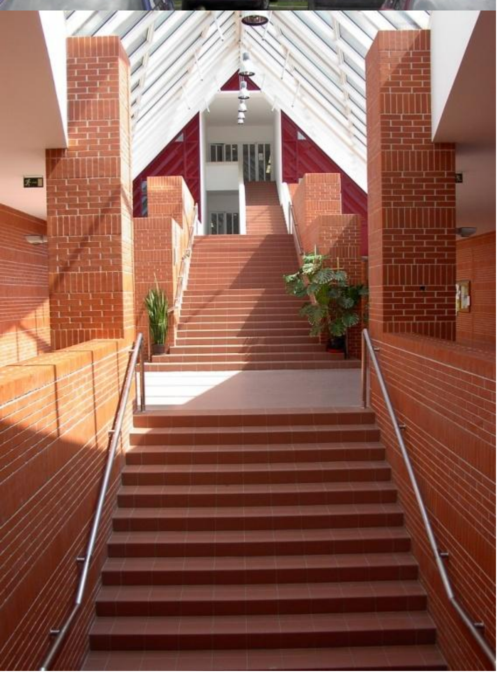
- Bemenet: $N,K \in \mathbb{N},\ X \in \mathbb{H}^N$
- Kimenet: $Van \in \mathbb{L},\ KI \in \mathbb{N}$
- Előfeltétel: $K>0$
- Utófeltétel: $Van = \exists i (1 \le i \le N): \sum_{\substack{j=1 \\ T(X_j)}}^{i} 1 = K$ és

  $Van \to 1 \le KI \le N$ és $\sum_{\substack{j=1 \\ T(X_j)}}^{KI} 1 = K$ és $T(X_{KI})$

Specifikáció:

- Bemenet: $N,K \in \mathbb{N}$, $X \in H^N$
- Kimenet: $Van \in \mathbb{L}$, $KI \in \mathbb{N}$
- Előfeltétel: $K>0$
- Utófeltétel: $Van = \exists i (1 \le i \le N): \sum_{\substack{j=1 \\ T(X_j)}}^{i} 1 = K$ és

$$Van \to 1 \le KI \le N \text{ és } \sum_{\substack{j=1 \\ T(X_j)}}^{KI} 1 = K \text{ és } T(X_{KI})$$



Keresés + megszámolás

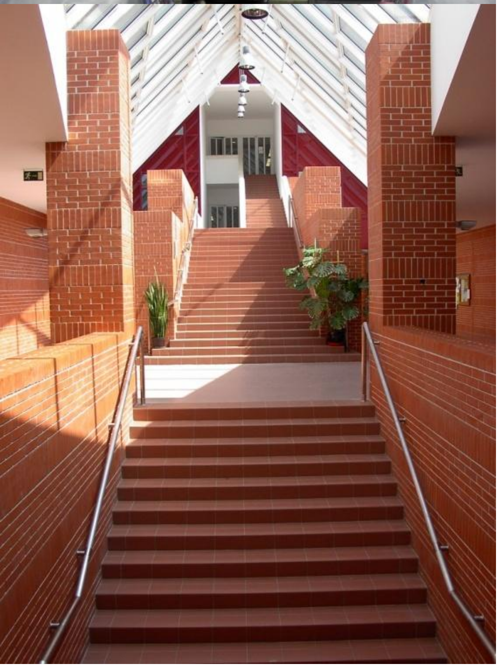
1. megoldási ötlet:

Az előbbi ötlet: „számoljuk meg, hogy hány adott tulajdonságú van, majd nézzük meg, hogy ez legalább K-e…” kevés, még hátra van a K. újbóli megkeresése…

Specifikáció:

- Bemenet: $N,K \in \mathbb{N}$, $X \in H^N$
- Kimenet: $Van \in \mathbb{L}$, $KI \in \mathbb{N}$
- Előfeltétel: $K>0$
- Utófeltétel: $Van = \exists i (1 \le i \le N): \sum_{\substack{j=1 \\ T(X_j)}}^{i} 1 = K$ és

  $Van \rightarrow 1 \le KI \le N$ és $\sum_{\substack{j=1 \\ T(X_j)}}^{KI} 1 = K$ és $T(X_{KI})$



Keresés + megszámolás

1. megoldási ötlet:

Az előbbi ötlet: „számoljuk meg, hogy hány adott tulajdonságú van, majd nézzük meg, hogy ez legalább K-e…” kevés, még hátra van a K. újbóli megkeresése…

A működőnek látszó ötlet: a megszámolás helyett kiválogatás kell… és a keresésre nincs szükség…
… de helypazarló és túl hosszadalmas!

Specifikáció:

- Bemenet: $N, K \in \mathbb{N}, X \in H^N$
- Kimenet: $Van \in \mathbb{L}, KI \in \mathbb{N}$
- Előfeltétel: $K>0$
- Utófeltétel: $Van = \exists i (1 \le i \le N): \sum_{\substack{j=1 \\ T(X_j)}}^{i} 1 = K$ és

  $Van \rightarrow 1 \le KI \le N$ és $\sum_{\substack{j=1 \\ T(X_j)}}^{KI} 1 = K$ és $T(X_{KI})$

```
i:=1
   i≤N és nem T(X[i])
      i:=i+1
Van:=i≤N
        Van
I                N
Ind:=i    |   —
```

```
Db:=0
   i=1..N
           T(X[i])
      I              N
      Db:=Db+1  |  —
```

Keresés + megszámolás

2. megoldási ötlet:

Ha már találtunk K darab adott tulajdonságút, akkor ne nézzük tovább: keresés a K.-ig.

Specifikáció:
- Bemenet: $N, K \in \mathbb{N}, X \in H^N$
- Kimenet: $Van \in \mathbb{L}, KI \in \mathbb{N}$
- Előfeltétel: $K>0$
- Utófeltétel: $Van = \exists i (1 \le i \le N): \sum_{\substack{j=1 \\ T(X_j)}}^{i} 1 = K$ és

  $Van \rightarrow 1 \le KI \le N$ és $\sum_{\substack{j=1 \\ T(X_j)}}^{KI} 1 = K$ és $T(X_{KI})$

Specifikáció:
- Bemenet: $N \in \mathbb{N}, X \in H^N$
- Kimenet: $Van \in \mathbb{L}, Ind \in \mathbb{N}$
- Előfeltétel: –
- Utófeltétel: $Van = \exists i\ (1 \le i \le N): T(X_i)$ és

  $Van \rightarrow 1 \le Ind \le N$ és $T(X_{Ind})$

Specifikáció:
- Bemenet: $N \in \mathbb{N}$,

  $X \in H^N$,

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

```
i:=1
i≤N és nem T(X[i])
    i:=i+1
Van:=i≤N
        Van
I                 N
Ind:=i   |   —
```

```
Db:=0
i=1..N
        T(X[i])
    I              N
    Db:=Db+1  |  —
```

Keresés + megszámolás

2. megoldási ötlet:

Ha már találtunk K darab adott tulajdonságút, akkor ne nézzük tovább: keresés a K.-ig.

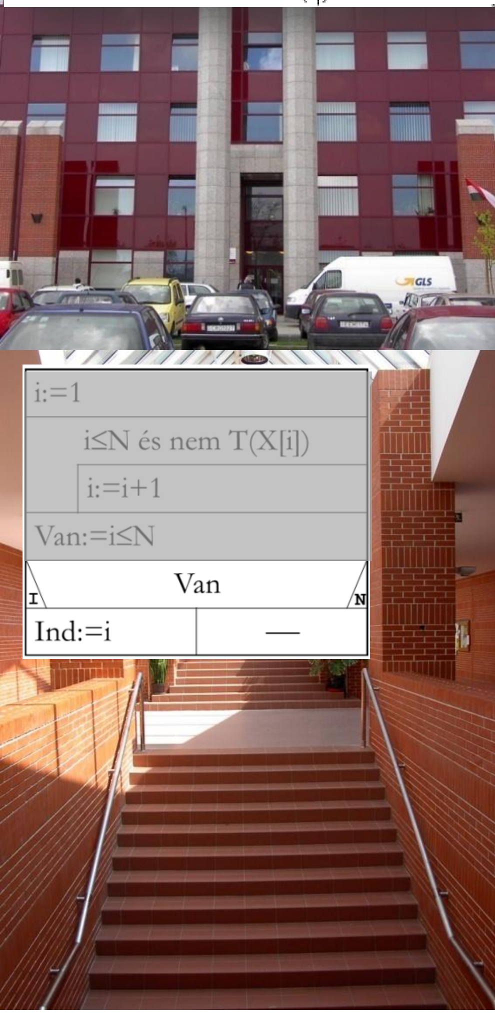
```
db:=0
i:=1
i≤N és db<K
        T(X[i])
    I                    N
    db:=db+1    |    —
    i:=i+1
Van:=db=K
...
```

Változó
 i:**Egész**

Specifikáció:

- Bemenet: $N, K \in \mathbb{N}, X \in H^N$
- Kimenet: $Van \in \mathbb{L}, KI \in \mathbb{N}$
- Előfeltétel: $K>0$
- Utófeltétel: $Van = \exists i(1 \le i \le N): \sum_{\substack{j=1 \\ T(X_j)}}^{i} 1 = K$ és

  $Van \rightarrow 1 \le KI \le N$ és $\sum_{\substack{j=1 \\ T(X_j)}}^{KI} 1 = K$ és $T(X_{KI})$



i:=1	
i≤N és nem T(X[i])	
i:=i+1	
Van:=i≤N	
Van	
Ind:=i	—

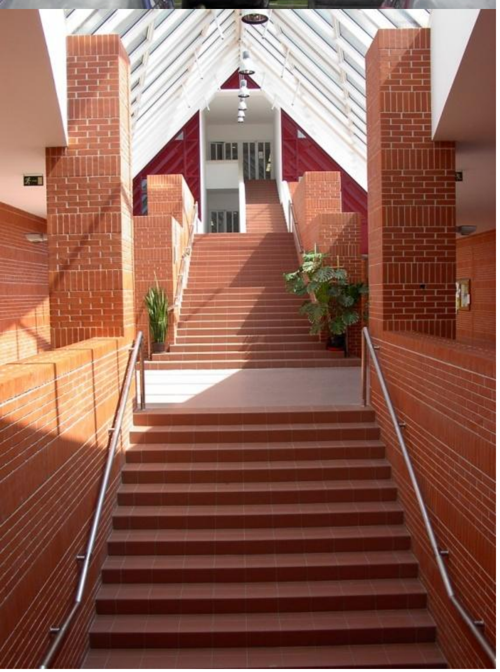
Keresés + megszámolás

2. megoldási ötlet:

Ha megtaláltunk a K.-at, akkor jegyezzük föl az indexét!

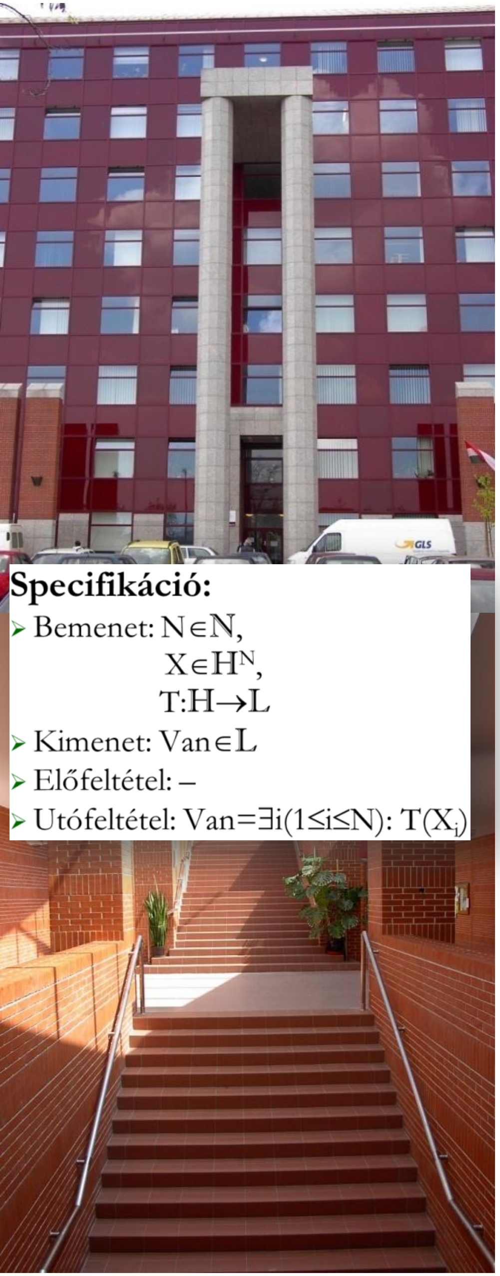
...	
Van	
KI:=i–1	

Eldöntés + eldöntés



Feladat:

Van-e két sorozatnak közös eleme?



Eldöntés + eldöntés

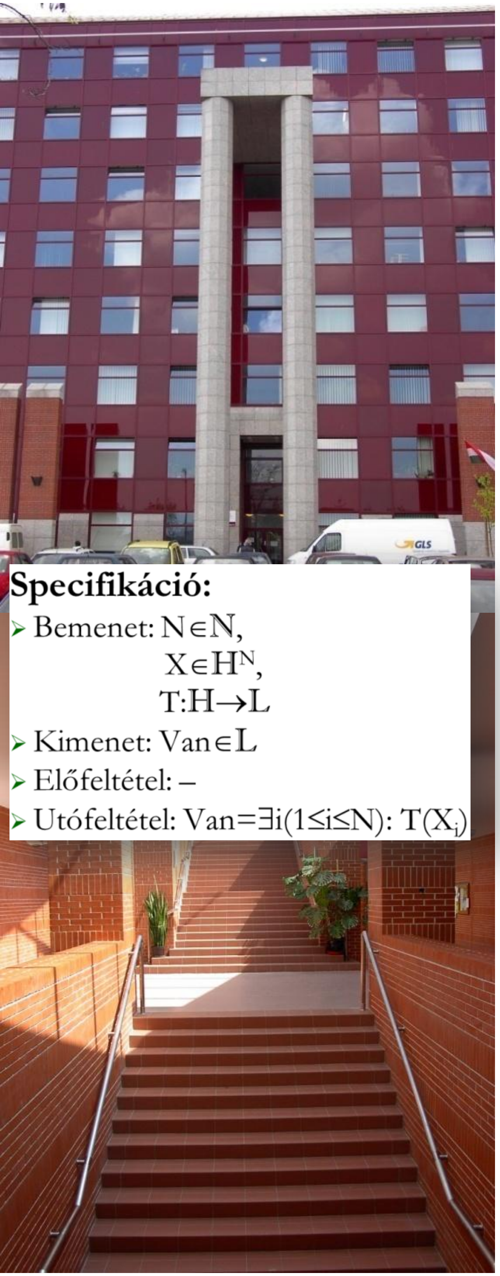
Feladat:

Van-e két sorozatnak közös eleme?

Specifikáció:

- Bemenet: $N,M \in \mathbb{N}$, $X \in \mathbb{H}^N$, $Y \in \mathbb{H}^M$
- Kimenet: $Van \in \mathbb{L}$
- Előfeltétel: –
- Utófeltétel: $Van = \exists i(1 \leq i \leq N), \exists j(1 \leq j \leq M): X_i = Y_j$

Eldöntés + eldöntés



Feladat:

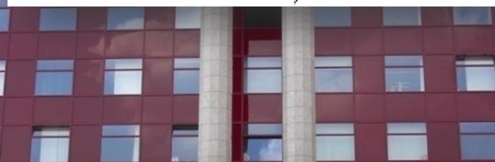
Van-e két sorozatnak közös eleme?

Specifikáció:

- Bemenet: $N,M \in \mathbb{N}$, $X \in \mathbb{H}^N$, $Y \in \mathbb{H}^M$
- Kimenet: $Van \in \mathbb{L}$
- Előfeltétel: –
- Utófeltétel: $Van = \exists i(1 \le i \le N), \exists j(1 \le j \le M): X_i = Y_j$
- Utófeltétel’: $Van = \underset{i=1}{\overset{N}{\exists}} \left(\underset{j=1}{\overset{M}{\exists}} X_i = Y_j \right)$

Specifikáció:

- Bemenet: $N,M\in N$, $X\in H^N$, $Y\in H^M$
- Kimenet: $Van\in L$
- Előfeltétel: –
- Utófeltétel: $Van=\exists i(1\leq i\leq N), \exists j(1\leq j\leq M): X_i=Y_j$



Specifikáció:

- Bemenet: $N,M\in N$, $X\in H^N$, $Y\in H^M$
- Kimenet: $Db\in N$, $Z\in H^{Db}$
- Előfeltétel: HalmazE(X) és HalmazE(Y)
- Utófeltétel: $Db=\sum_{\substack{i=1\\X_i\in Y}}^{N} 1$ és $\forall i(1\leq i\leq Db): (Z_i\in X$ és $Z_i\in Y)$ és HalmazE(Z)

Eldöntés + eldöntés

1. megoldási ötlet:

Határozzuk meg a két sorozat közös elemeit (metszet), s ha ennek elemszáma legalább 1, akkor van közös elem!

Specifikáció:

- Bemenet: $N,M \in \mathbb{N}$, $X \in H^N$, $Y \in H^M$
- Kimenet: $Van \in \mathbb{L}$
- Előfeltétel: –
- Utófeltétel: $Van = \exists i(1 \le i \le N), \exists j(1 \le j \le M): X_i = Y_j$

Specifikáció:

- Bemenet: $N,M \in \mathbb{N}$, $X \in H^N$, $Y \in H^M$
- Kimenet: $Db \in \mathbb{N}$, $Z \in H^{Db}$
- Előfeltétel: HalmazE(X) és HalmazE(Y)
- Utófeltétel: $Db = \sum_{\substack{i=1 \\ X_i \in Y}}^{N} 1$ és $\forall i(1 \le i \le Db): (Z_i \in X$ és $Z_i \in Y)$ és HalmazE(Z)

Eldöntés + eldöntés

1. megoldási ötlet:

Határozzuk meg a két sorozat közös elemeit (metszet), s ha ennek elemszáma legalább 1, akkor van közös elem!

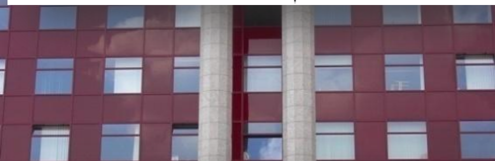
Specifikáció:

- Az utófeltétel „igazítása”:
  - a metszet részeredménye volt: $Db \in \mathbb{N}$
  - a módosított utófeltétel:

    metszet utófeltétele és $Van = Db > 0$.

Specifikáció:

- Bemenet: $N,M \in \mathbb{N}$, $X \in H^N$, $Y \in H^M$
- Kimenet: $Van \in \mathbb{L}$
- Előfeltétel: –
- Utófeltétel: $Van = \exists i(1 \le i \le N), \exists j(1 \le j \le M): X_i = Y_j$



Specifikáció:

- Bemenet: $N,M \in \mathbb{N}$, $X \in H^N$, $Y \in H^M$
- Kimenet: $Db \in \mathbb{N}$, $Z \in H^{Db}$
- Előfeltétel: HalmazE(X) és HalmazE(Y)
- Utófeltétel: $Db = \sum_{\substack{i=1 \\ X_i \in Y}}^{N} 1$ és $\forall i(1 \le i \le Db)$: ($Z_i \in X$ és $Z_i \in Y$) és HalmazE(Z)

Eldöntés + eldöntés

1. megoldási ötlet:

Határozzuk meg a két sorozat közös elemeit (metszet), s ha ennek elemszáma legalább 1, akkor van közös elem!

Specifikáció:

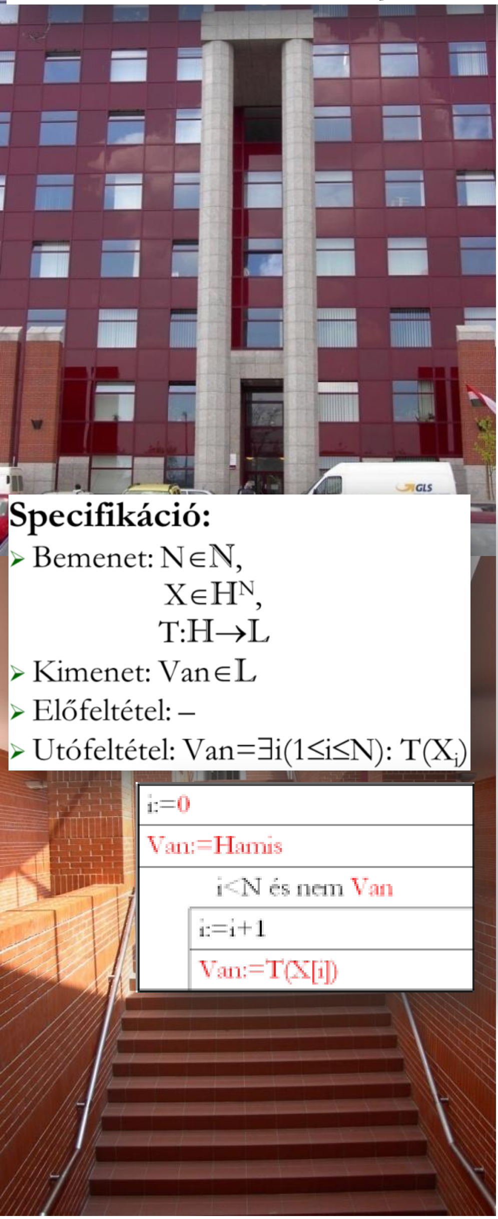
- Az utófeltétel „igazítása”:
  - a metszet részeredménye volt: $Db \in \mathbb{N}$
  - a módosított utófeltétel:

    metszet utófeltétele és $Van = Db > 0$.

Megjegyzés:

A metszet = kiválogatás + eldöntés

- Utófeltétel': $Van = \exists_{i=1}^{N}\left(\exists_{j=1}^{M} X_i = Y_j\right)$



Specifikáció:

- Bemenet: $N \in \mathbb{N}$,
  $X \in H^N$,
  $T: H \rightarrow L$
- Kimenet: $Van \in L$
- Előfeltétel: –
- Utófeltétel: $Van = \exists i(1 \le i \le N): T(X_i)$

i:=0
Van:=Hamis
i<N és nem Van
i:=i+1
Van:=T(X[i])

Eldöntés + eldöntés

2. megoldási ötlet :

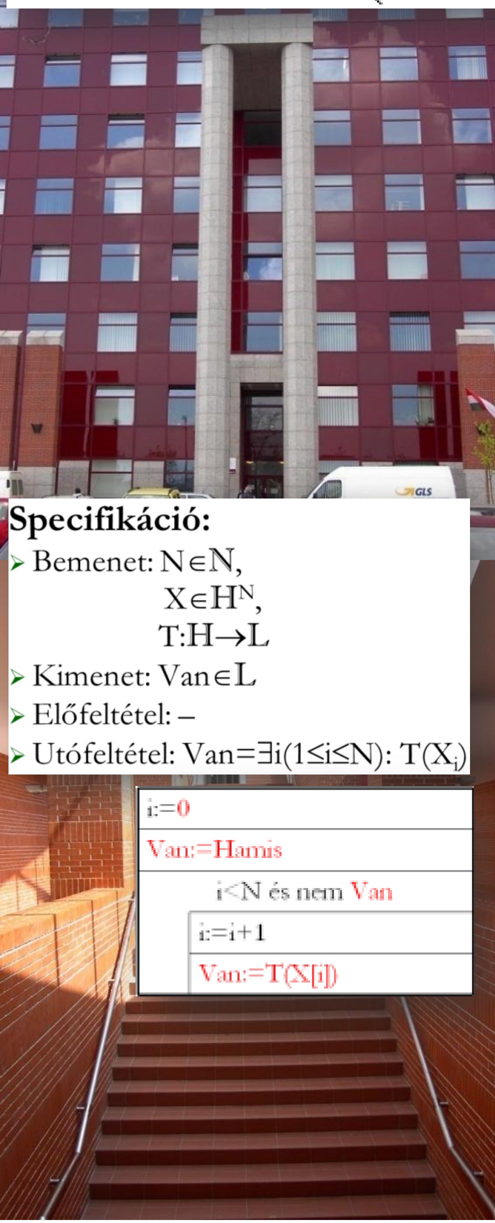
Ha már találtunk 1 darab közös elemet, akkor ne nézzük tovább!

Változó
i,j:**Egész**

i:=0
Van:=Hamis
i<N és nem Van
i:=i+1
j:=1
j≤M és X[i]≠Y[j]
j:=j+1
Van:=j≤M

- Utófeltétel': $Van = \exists_{i=1}^{N} \left(\exists_{j=1}^{M} X_i = Y_j \right)$

Eldöntés + eldöntés



Specifikáció:

- Bemenet: $N \in \mathbb{N}$,
  $X \in H^N$,
  $T: H \to \mathbb{L}$
- Kimenet: $Van \in \mathbb{L}$
- Előfeltétel: –
- Utófeltétel: $Van = \exists i (1 \le i \le N): T(X_i)$

```
i:=0
Van:=Hamis
     i<N és nem Van
   i:=i+1
   Van:=T(X[i])
```

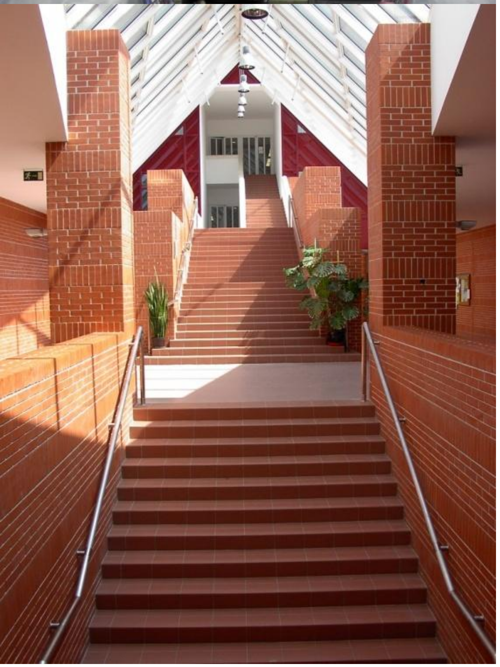
2. megoldási ötlet :

Ha már találtunk 1 darab közös elemet, akkor ne nézzük tovább!

Változó
i,j:**Egész**

```
i:=0
Van:=Hamis
          i<N és nem Van
     i:=i+1
     j:=1
            j≤M és X[i]≠Y[j]
          j:=j+1
     Van:=j≤M
```

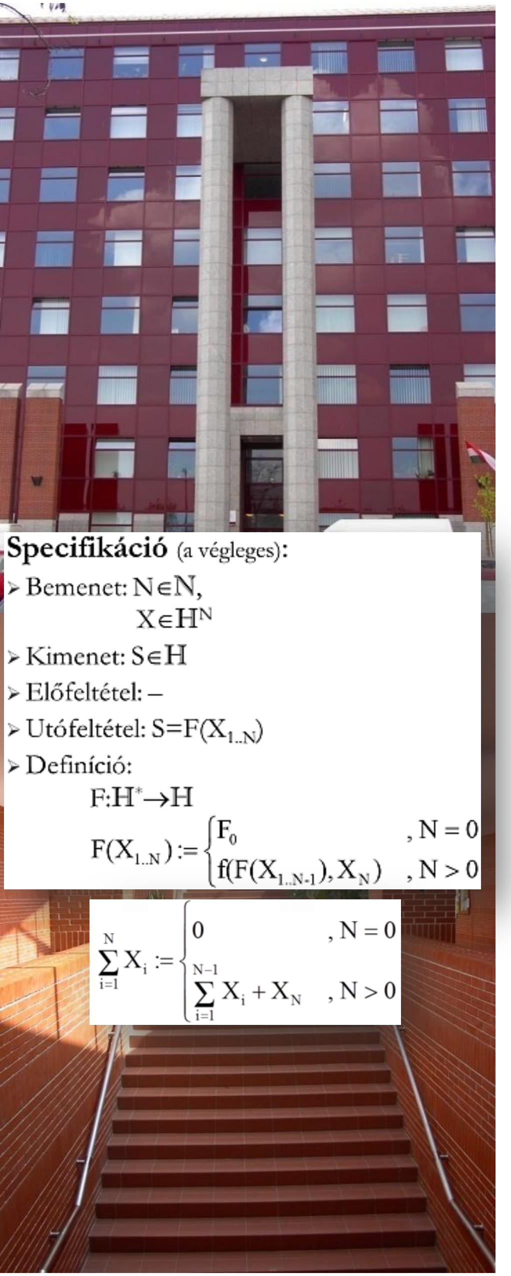
Összegzés mátrixra



Feladat:

Egy mátrix elemeinek összege.

Összegzés mátrixra



Specifikáció (a végleges):

- Bemenet: $N \in \mathbb{N}$, $X \in H^N$
- Kimenet: $S \in H$
- Előfeltétel: –
- Utófeltétel: $S = F(X_{1..N})$
- Definíció:

$$F: H^* \to H$$

$$F(X_{1..N}) := \begin{cases} F_0 & , N = 0 \\ f(F(X_{1..N-1}), X_N) & , N > 0 \end{cases}$$

$$\sum_{i=1}^{N} X_i := \begin{cases} 0 & , N = 0 \\ \sum_{i=1}^{N-1} X_i + X_N & , N > 0 \end{cases}$$

Feladat:

Egy mátrix elemeinek összege.

Specifikáció:

- Bemenet: $N, M \in \mathbb{N}$, $X \in \mathbb{Z}^{N \times M}$
- Kimenet: $S \in \mathbb{Z}$
- Előfeltétel: –
- Utófeltétel: $S = \sum_{i=1}^{N} \left(\sum_{j=1}^{M} X_{i,j} \right)$

Összegzés mátrixra

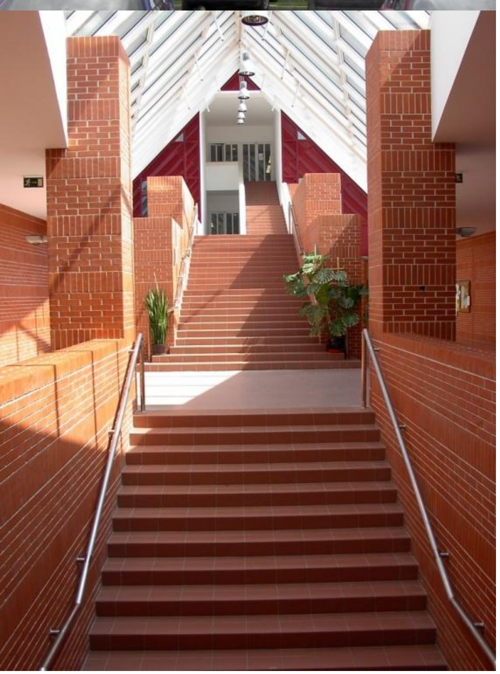
Algoritmus:

A megoldás lényegében csak abban különbözik az alapváltozattól, hogy a mátrix miatt két –egymásba ágyazott– ciklusra van szükség.

Változó
i,j:**Egész**

```
S:=0
i=1..N
    j=1..M
        S:=S+X[i,j]
```

Specifikáció:
- Bemenet: $N,M \in \mathbb{N}$, $X \in \mathbb{Z}^{N \times M}$
- Kimenet: $S \in \mathbb{Z}$
- Előfeltétel: –
- Utófeltétel: $S=\sum_{i=1}^{N}\left(\sum_{j=1}^{M} X_{i,j}\right)$



Eldöntés mátrixra

Feladat:

Van-e egy mátrixban adott tulajdonságú elem?

Eldöntés mátrixra

Feladat:

Van-e egy mátrixban adott tulajdonságú elem?

Specifikáció:

- Bemenet: $N,M \in \mathbb{N}$, $X \in \mathbb{H}^{N \times M}$
- Kimenet: $Van \in \mathbb{L}$
- Előfeltétel: –
- Utófeltétel: $Van = \exists i(1 \leq i \leq N), \exists j(1 \leq j \leq M): T(X_{i,j})$

Eldöntés mátrixra

Feladat:

Van-e egy mátrixban adott tulajdonságú elem?

Specifikáció:

- Bemenet: $N,M \in \mathbb{N}$, $X \in \mathbb{H}^{N \times M}$
- Kimenet: $Van \in \mathbb{L}$
- Előfeltétel: –
- Utófeltétel: $Van = \exists i(1 \le i \le N), \exists j(1 \le j \le M): T(X_{i,j})$
- Utófeltétel’: $Van = \overset{N}{\underset{i=1}{\exists}} \left(\overset{M}{\underset{j=1}{\exists}} T(X_{i,j}) \right)$

i:=1

i≤N és nem T(X[i])

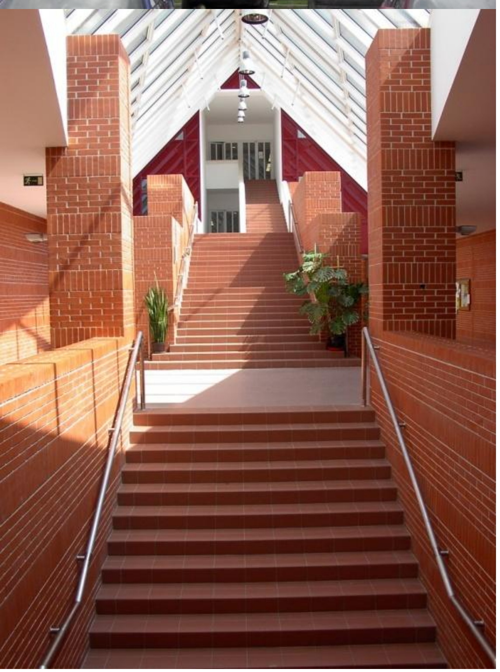
i:=i+1

Van:= i≤N

Eldöntés mátrixra

Algoritmus:

Az alapváltozathoz képest itt meg kell fogalmazni a mátrix elemein való –nem feltétlenül– végighaladást, soronként, balról jobbra!



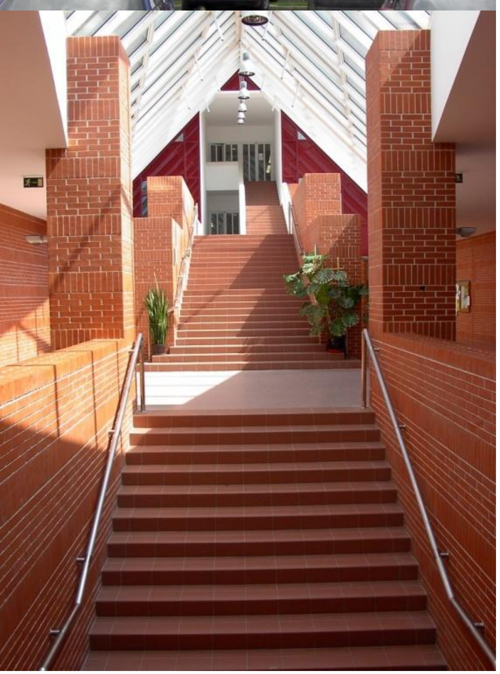
i:=1

iN és nem T(X[i])

i:=i+1

Van:= i≤N

Eldöntés mátrixra



Algoritmus:

Az alapváltozathoz képest itt meg kell fogalmazni a mátrix elemein való –nem feltétlenül– végighaladást, soronként, balról jobbra!

Változó
i,j:**Egész**

i:=1

j:=1

i≤N és nem T(X[i,j])

j<M

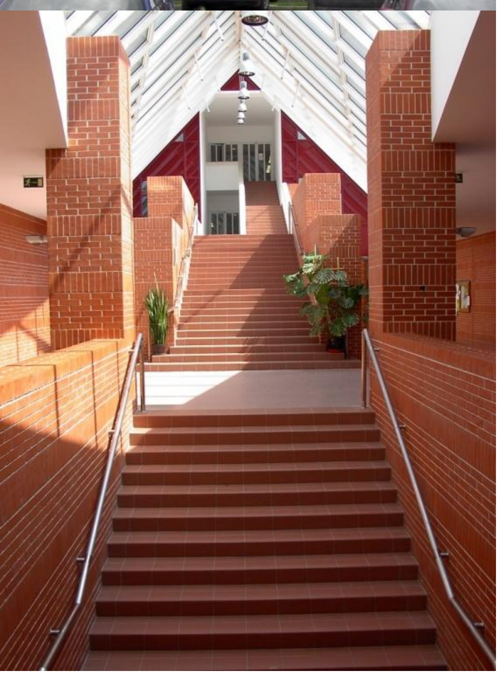
I | N

j:=j+1 | j:=1, i:=i+1

Van:= i≤N

Eldöntés mátrixra

i:=1
i≤N és nem T(X[i])
i:=i+1
Van:= i≤N



Algoritmus:

Az alapváltozathoz képest itt meg kell fogalmazni a mátrix elemein való –nem feltétlenül– végighaladást, soronként, balról jobbra!

i:=1
j:=1
i≤N és nem T(X[i,j])
j<M (I / N)
I: j:=j+1
N: j:=1 ; i:=i+1
Van:= i≤N

Változó
i,j:**Egész**

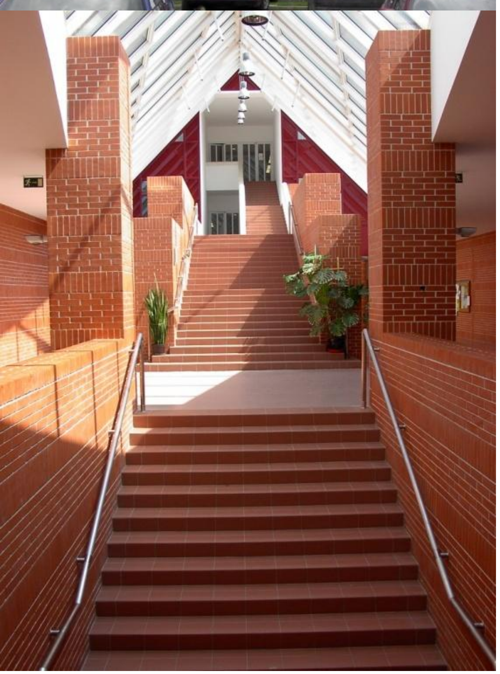
Eldöntés mátrixra

i:=1

$\boxed{i\leq N}$ és nem T(X[i])

  i:=i+1

Van:= i≤N



Algoritmus:

Az alapváltozathoz képest itt meg kell fogalmazni a mátrix elemein való –nem feltétlenül– végighaladást, soronként, balról jobbra!

Változó
  i,j:**Egész**

i:=1

j:=1

$\boxed{i\leq N}$ és nem T(X[i,j])

  j<M (I / N)

  I: j:=j+1

  N: j:=1; i:=i+1

Van:= i≤N

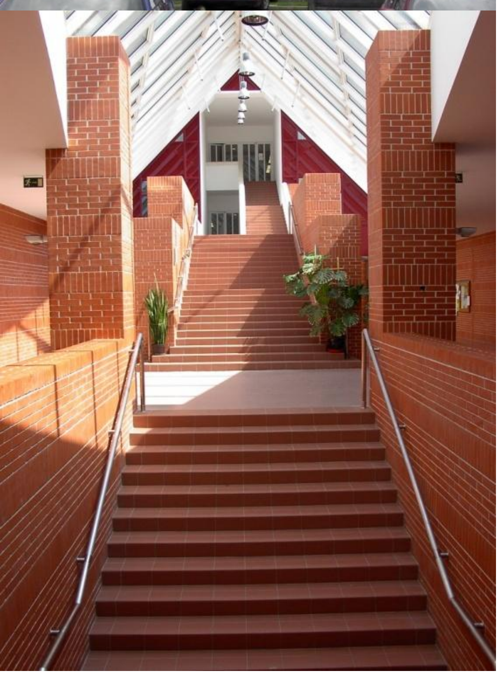
i:=1

i≤N és nem T(X[i])

i:=i+1

Van:= i≤N

Eldöntés mátrixra



Algoritmus:

Az alapváltozathoz képest itt meg kell fogalmazni a mátrix elemein való –nem feltétlenül– végighaladást, soronként, balról jobbra!

Változó
i,j:**Egész**

i:=1

j:=1

i≤N és nem T(X[i,j])

j<M

I	N
j:=j+1	j:=1 i:=i+1

Van:= i≤N

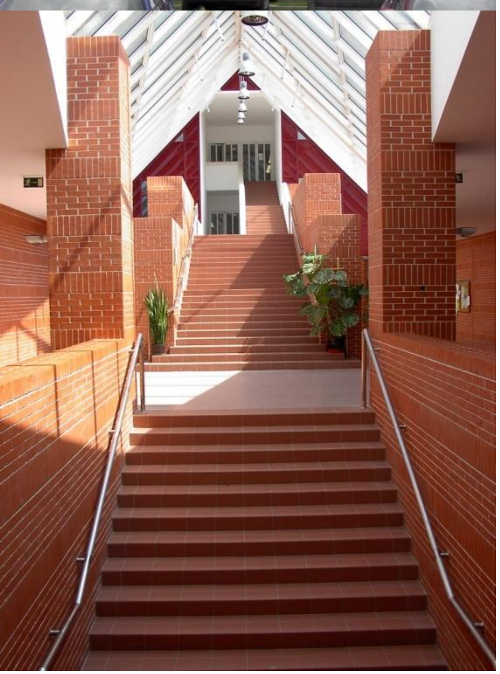
Eldöntés mátrixra

i:=1

i≤N és nem T(X[i])

i:=i+1

Van:= i≤N



Algoritmus:

Az alapváltozathoz képest itt meg kell fogalmazni a mátrix elemein való –nem feltétlenül– végighaladást, soronként, balról jobbra!

Változó
i,j:**Egész**

i:=1

j:=1

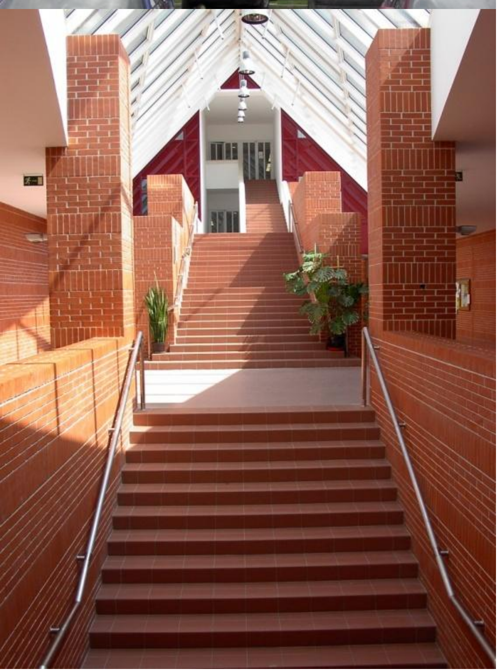
i≤N és nem T(X[i,j])

j<M

I: j:=j+1

N: j:=1; i:=i+1

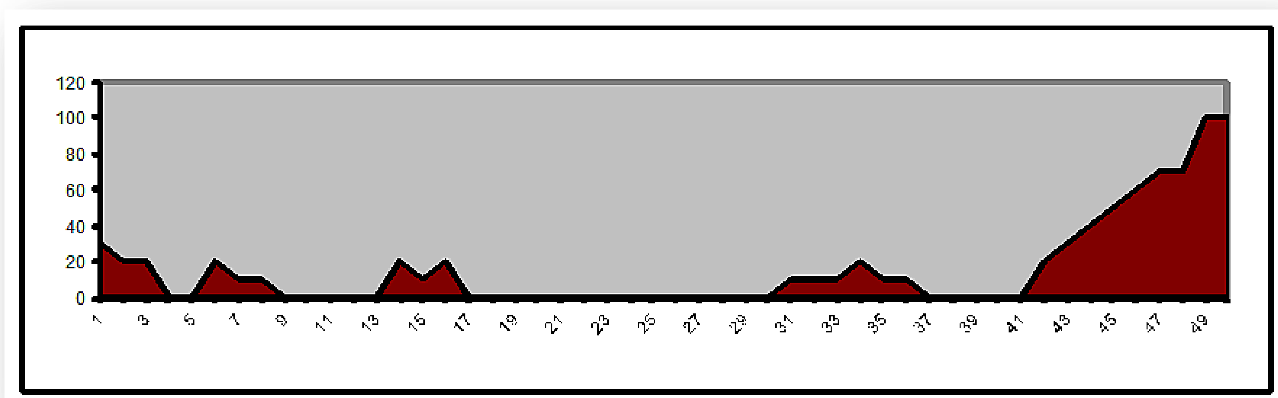
Van:= i≤N

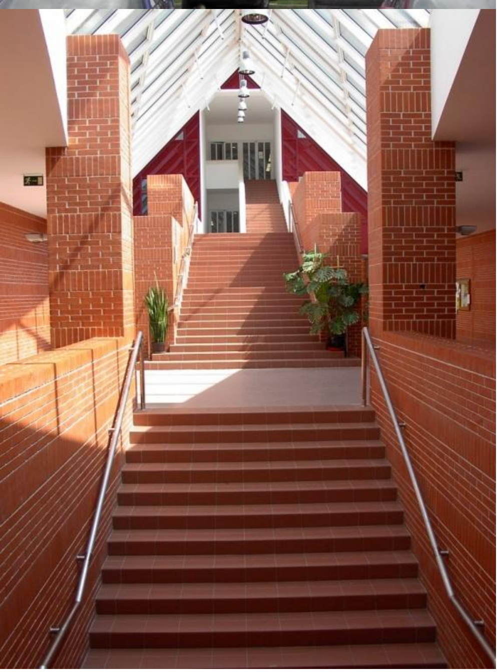


Tesztek előállítása

Feladat (teszteléshez)**:**

Egy repülőgéppel Európából Amerikába repültünk. Az út során X kilométerenként mértük a felszín tengerszint feletti magasságát (≥0). 0 magasságot ott mértünk, ahol tenger van, >0-t pedig ott, ahol szárazföld. Adjuk meg a szigeteket!

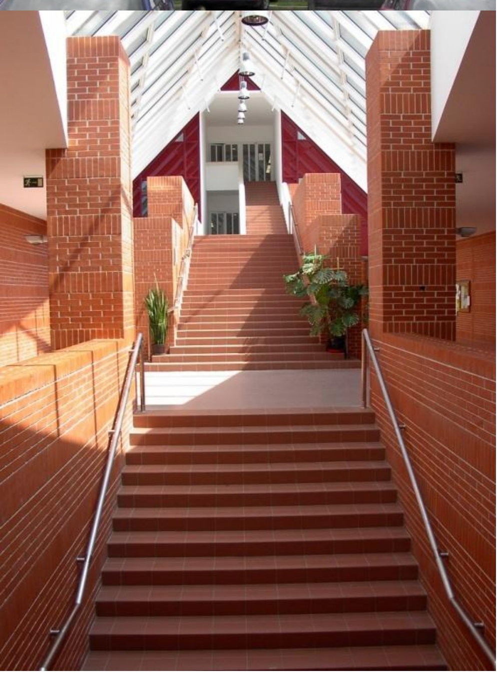




Tesztek előállítása

Specifikáció:

- Bemenet: $N \in \mathbb{N}$, $Mag \in \mathbb{N}^N$
- Kimenet: $Db \in \mathbb{N}$, $K,V \in \mathbb{N}^{Db}$
- …



Tesztek előállítása

Specifikáció:

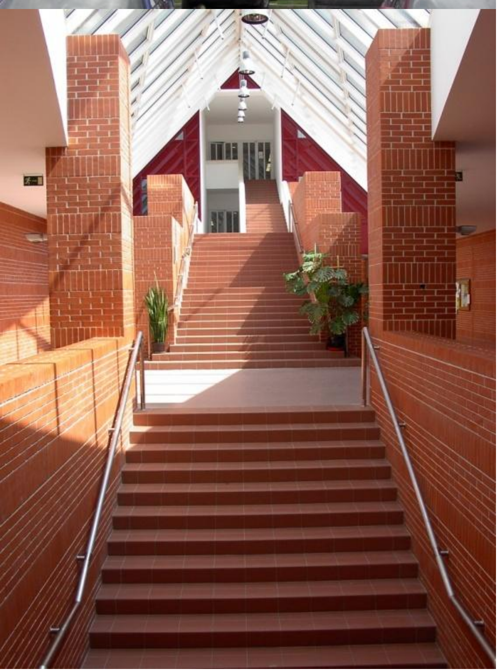
- Bemenet: $N\in\mathbb{N}$, $Mag\in\mathbb{N}^N$
- Kimenet: $Db\in\mathbb{N}$, $K,V\in\mathbb{N}^{Db}$
- …

Tesztelés:

- **Kis** tesztek a tesztelési elveknek megfelelően, például:

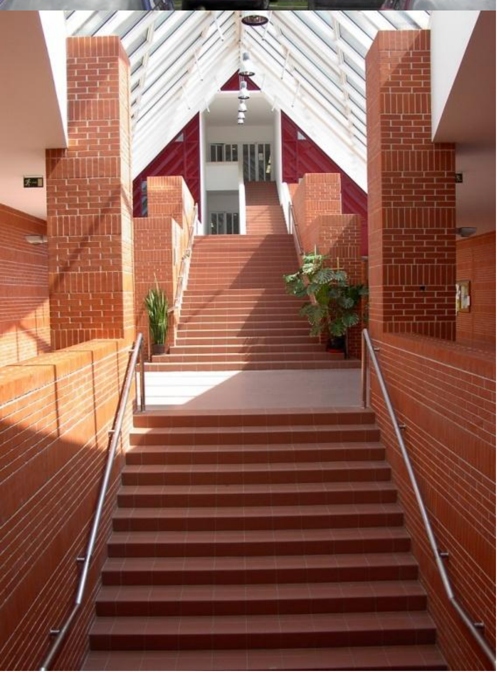
  N=3, Mag=(1,0,1) → nincs sziget
  N=5, Mag=(1,0,1,0,1) → egy „rövid" sziget
  N=7, Mag=(1,0,1,0,1,0,1) → több „rövid" sziget
  N=7, Mag=(1,0,1,1,1,0,1) → hosszabb sziget

- Hogyan készítünk **nagy** (hatékonysági) teszteket?



Szabályos tesztek

- Generálhatunk „**szabályos**” teszteket (egyszerű ciklusokkal). Például így:



Szabályos tesztek

- Generálhatunk „**szabályos**” teszteket (egyszerű ciklusokkal). Például így:

Változó
i:**Egész**

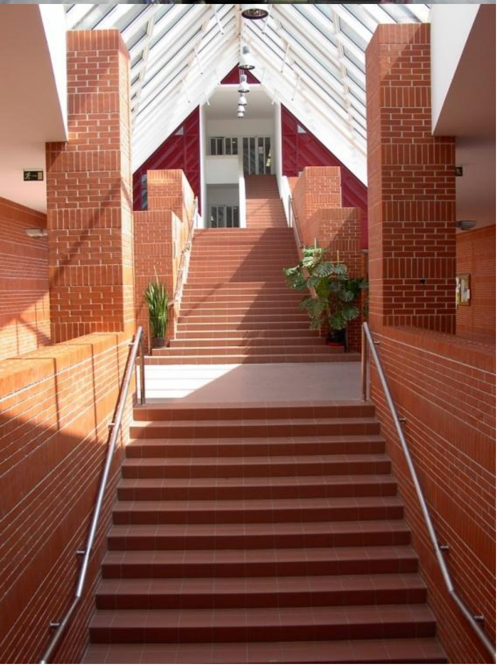
```
N:=1000
i=1..10
    Mag[i]:=11–i
i=11..900
    Mag[i]:=0
i=901..N
    Mag[i]:=i–900
```

Szabályos tesztek

- Generálhatunk „**szabályos**” teszteket (egyszerű ciklusokkal). Például így:

Változó
i:**Egész**

```
N:=1000
i=1..10
    Mag[i]:=11–i          ⇐ Európa
i=11..900
    Mag[i]:=0             ⇐ tenger
i=901..N
    Mag[i]:=i–900         ⇐ Amerika
```



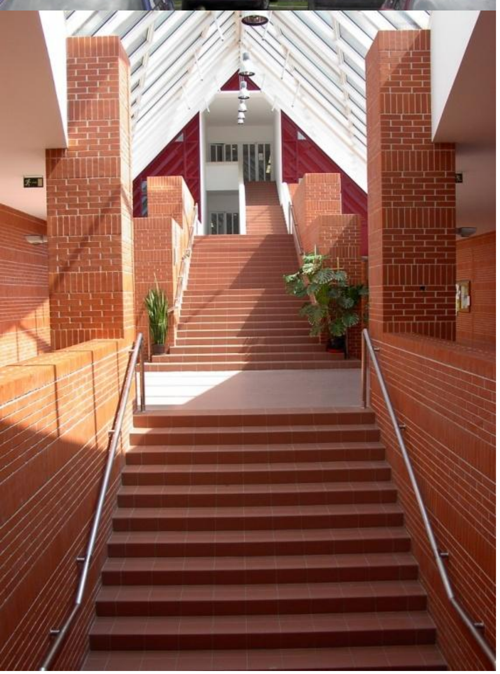
Véletlen tesztek
(alapok – véletlenszámok)

- A **véletlenszám**okat a számítógép egy algoritmussal állítja elő egy kezdőszámból kiindulva.

  $x_0 \rightarrow f(x_0)=x_1 \rightarrow f(x_1)=x_2 \rightarrow \ldots$
- A „véletlenszerűséghez” megfelelő függvény és jó kezdőszám szükséges.
- **Kezdőszám**: (pl.) a belső órából vett érték.
- **Függvény** (az ún. lineáris kongruencia módszernél):

  $f(x) = (A*x+B) \text{ Mod } M$,

  ahol A, B és M a függvény belső konstansai.



Véletlen tesztek
(alapok – C++)

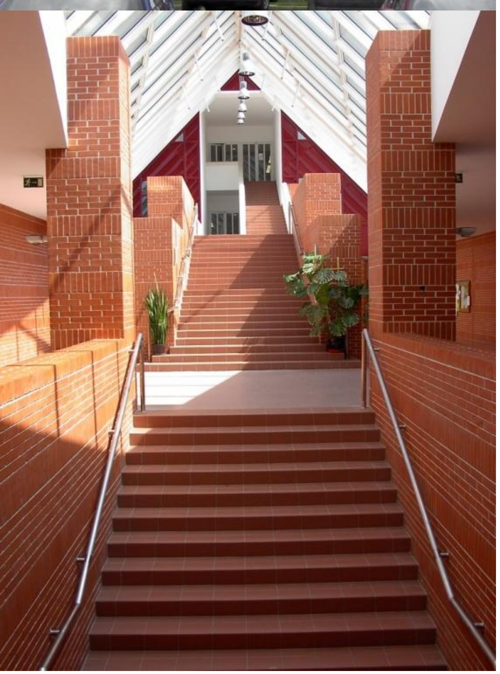
C++: rand() véletlen egész számot ad 0 és egy maximális érték (RAND_MAX) között. srand(szám) kezdőértéket állít be.

- Véletlen(a..b)∈{a,…,b} `v=rand() % (b-a+1)+a`
- Véletlen(N)∈{1,…,N} `v=rand() % N+1`
- véletlenszám∈[0,1)⊂ℝ

```
v=rand()/(RAND_MAX+1.0)
```

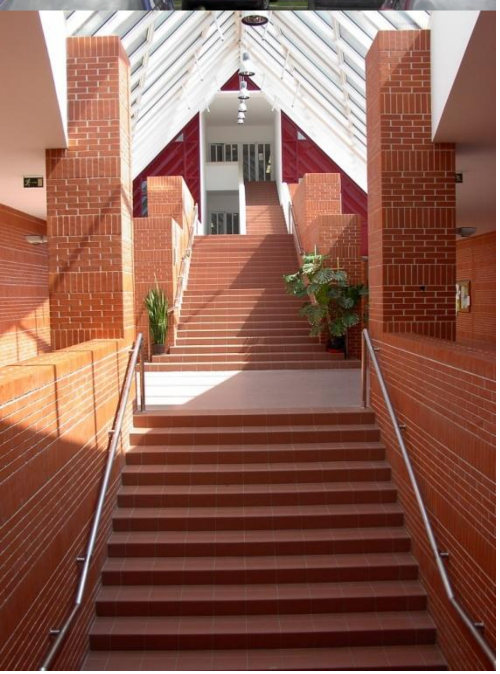
- A generátor használata kockadobásra:

```
#include <time.h>
…
srand(time(NULL));
i=rand() % 6 +1;
```



Véletlen tesztek

- **Véletlen** tesztekhez használjunk véletlenszámokat! Például így:

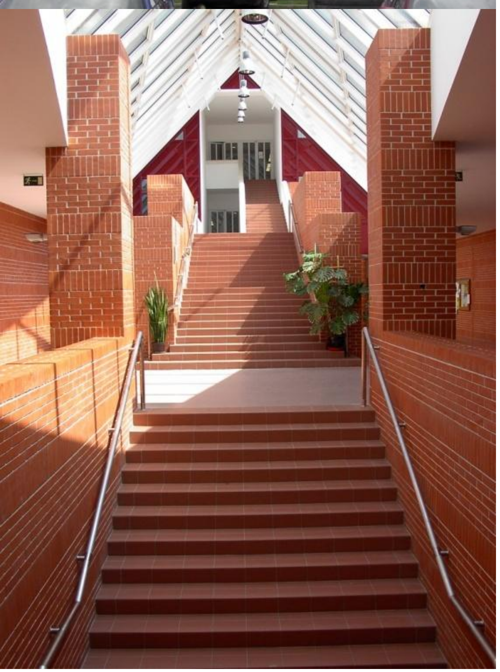


Véletlen tesztek

- **Véletlen** tesztekhez használjunk véletlenszá-mokat! Például így:

Változó
i:**Egész**

N:=1000
M:=Véletlen(10)
i=1..M
Mag[i]:=Véletlen(5..10)
i=M+1..900
véletlenszám<0.5 (I / N)
I: Mag[i]:=0 — N: Mag[i]:=1
i=901..N
Mag[i]:=Véletlen(3..8)



Véletlen tesztek

- **Véletlen** tesztekhez használjunk véletlenszámokat! Például így:

Változó
i:**Egész**

```
N:=1000
M:=Véletlen(10)
i=1..M
    Mag[i]:=Véletlen(5..10)          ⇐ Európa
i=M+1..900
    véletlenszám<0.5                 ⇐ tenger és szigetek
    I: Mag[i]:=0   |   N: Mag[i]:=1
i=901..N
    Mag[i]:=Véletlen(3..8)           ⇐ Amerika
```

Véletlen tesztek
(Példa – C++)

Specifikációs komment:
Az előbbi algoritmus általánosítása!

Fájlkezeléshez

Kód:

```
//név: Gipsz Jakab
//ETR-azonosító: GIJAAFT.ELTE
//drótposta-cím: Gibs@elte.hu
//Feladat:
//  Véletlen tesztadatok generálása fájlba, a "tengeres" feladatokhoz.
//Specifikáció:
//  Be: N ELEME EGÉSZ [tesztadatok száma]
//      NEu,NAm ELEME EGÉSZ [Európára, Amerikára esõ mérési adatok "várható" száma]
//      PSzig ELEME VALÓS [szigetre esés valószínûsége]
//      MaxMag ELEME EGÉSZ [a generálható legnagyobb magasság]
//  Ki: Mag ELEME TÖMB[1..N:EGÉSZ]
//  Ef: NEu,NAm>=1 ÉS N>=NEu+NAm ÉS
//      PSzig ELEME [0..1) ÉS
//      MaxMag>0
//  Uf: Mag[1],Mag[N] ELEME [1..MaxMag] ÉS
//      LÉTEZIK eu ELEME [1..NEu]: LÉTEZIK am ELEME [1..NAm]:
//         BÁRMELY i ELEME [1..eu]: Mag[i] ELEME [1..MaxMag] ÉS Mag[eu+1]=0 ÉS
//         BÁRMELY i ELEME [N-am+1..N]: Mag[i] ELEME [1..MaxMag] ÉS Mag[am-1]=0 ÉS
//         BÁRMELY i ELEME [eu+1..N-am]: Mag[i] ELEME [0..MaxMag]
//  Megjegyzés: nem foglalkoztunk valószínûségi elvárásokkal!

#include <iostream>
#include <fstream>
//#include <time.h>//csak a 'srand(time(NULL))' rand()-inicializáláshoz kell!
#include <stdlib.h>//Code::Blocks 10.05-höz már kell a system kedvéért

using namespace std;
```

Véletlen tesztek

(Példa – C++)

Kód

```
//név: Gipsz Jakab
//ETR-azonosító: GIJAAFT.ELTE
//drótposta-cím: Gibs@elte.hu
//Feladat:
//  Véletlen tesztadatok generálása fájlba, a "tengeres" feladatokhoz.
//Specifikáció:
//  Be: N ELEME EGÉSZ [tesztadatok száma]
//      NEu,NAm ELEME EGÉSZ [Európára, Amerikára esõ mérési adatok "várható" szá
//      PSzig ELEME VALÓS [szigetre esés valószínûsége]
//      MaxMag ELEME EGÉSZ [a generálható legnagyobb magasság]
//  Ki: Mag ELEME TÖMB[1..N:EGÉSZ]
//  Ef: NEu,NAm>=1 ÉS N>=NEu+NAm ÉS
//      PSzig ELEME [0..1) ÉS
//      MaxMag>0
//  Uf: Mag[1],Mag[N] ELEME [1..MaxMag] ÉS
//      LÉTEZIK eu ELEME [1..NEu]: LÉTEZIK am ELEME [1..NAm]:
//         BÁRMELY i ELEME [1..eu]: Mag[i] ELEME [1..MaxMag] ÉS Mag[eu+1]=0 ÉS
//         BÁRMELY i ELEME [N-am+1..N]: Mag[i] ELEME [1..MaxMag] ÉS Mag[am-1]=0
//         BÁRMELY i ELEME [eu+1..N-am]: Mag[i] ELEME [0..MaxMag]
//  Megjegyzés: nem foglalkoztunk valószínûségi elvárásokkal!

#include <iostream>
#include <fstream>
//#include <time.h>//csak a 'srand(time(NULL))' rand()-inicializáláshoz kell!
#include <stdlib.h>//Code::Blocks 10.05-höz már kell a system kedvéért
```

Véletlen tesztek
(Példa – C++)

Függvény-prototípusok

Főprogram

Kód:

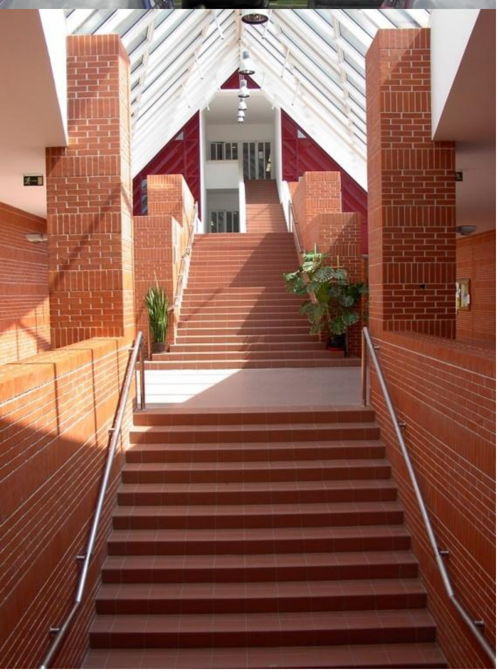
```
//név: Gipsz Jakab
```

```
//beolvassa a mnN..mxN közötti egész számot (mxN<mnN => végtelen)
void be_int(string kerd, int &n, int mnN, int mxN, string uz);
//beolvassa a mnN..mxN közötti valós számot (mxN<mnN => végtelen)
void be_float(string kerd, float &x, float mnN, int mxN, string uz);
//Fájlba generálás:
void fajlbaGeneralas(string fN, int n, int nEu, int nAm, float pT, int mxM);
void billentyureVar();

int main()
{
    //Bemenet:
    int N,NEu,NAm,MaxMag;
    float PSzig;
    //Kimenet:
    const string fN="tenger.csv";//kimeneti fájl neve
    //srand(time(NULL));//rand()-inicializáláshoz kell!
    //adatok beolvasása:
    be_int("Europai pontok (varhato) szama",NEu,1,0,"Hibas termeszetes szam!");
    be_int("Amerikai pontok (varhato) szama",NAm,1,0,"Hibas termeszetes szam!");
    be_int("Meresi pontok szama",N,NEu+NAm,0,"Hibas termeszetes szam!");
    be_float("Tengeri pontok valoszinusege",PSzig,0,1,"Hibas valos szam!");
    be_int("Legnagyobb magassag",MaxMag,1,0,"Hibas termeszetes szam!");
    //a lényeg:
    fajlbaGeneralas(fN,N,NEu,NAm,PSzig,MaxMag);
    billentyureVar();
    return 0;
}
```

Véletlen tesztek
(Példa – C++)

```
//beolvassa a mnN..mxN közötti egész számot (mxN<mnN => végtelen)
void be_int(string kerd, int &n, int mnN, int mxN, string uz);
//beolvassa a mnN..mxN közötti valós számot (mxN<mnN => végtelen)
void be_float(string kerd, float &x, float mnN, int mxN, string uz);
//Fájlba generálás:
void fajlbaGeneralas(string fN, int n, int nEu, int nAm, float pT, int mxM);
void billentyureVar();

int main()
{
    //Bemenet:
    int N,NEu,NAm,MaxMag;
    float PSzig;
    //Kimenet:
    const string fN="tenger.csv";//kimeneti fájl neve
    //srand(time(NULL));//rand()-inicializáláshoz kell!
    //adatok beolvasása:
    be_int("Europai pontok (varhato) szama",NEu,1,0,"Hibas termeszetes szam!");
    be_int("Amerikai pontok (varhato) szama",NAm,1,0,"Hibas termeszetes szam!");
    be_int("Meresi pontok szama",N,NEu+NAm,0,"Hibas termeszetes szam!");
    be_float("Tengeri pontok valoszinusege",PSzig,0,1,"Hibas valos szam!");
    be_int("Legnagyobb magassag",MaxMag,1,0,"Hibas termeszetes szam!");
    //a lényeg:
    fajlbaGeneralas(fN,N,NEu,NAm,PSzig,MaxMag);
    billentyureVar();
    return 0;
}
```

Véletlen tesztek
(Példa – C++)



A lényegi eljárás

Kód:

```
//név: Gipsz Jakab
```

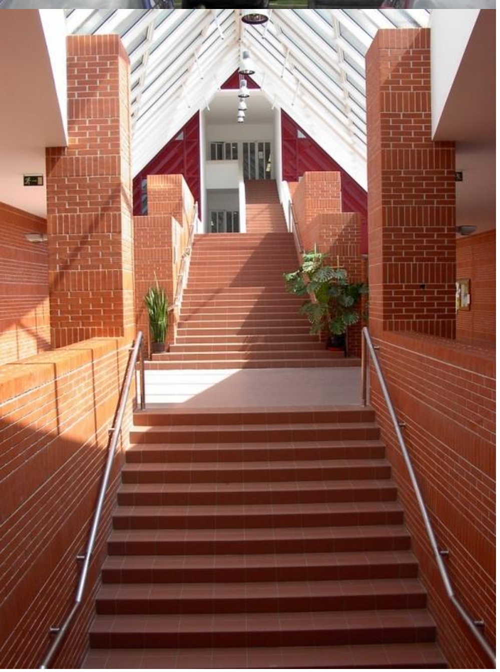
```
//beolvassa a mnN..mxN közötti egész számot (mxN<mnN => végtelen)
```

```
//Fájlba generálás:
void fajlbaGeneralas(string fN, int n, int nEu, int nAm, float pT, int mxM)
//Uf: N sorban, soronként egyetlen 0..mxM közötti egész szám ...
{
    ofstream oF(fN.c_str());
    int eu=rand()%nEu+1;//Európa hossza
    int am=rand()%nAm+1;//Amerika hossza
    for(int i=1;i<=eu;++i) {
      oF << rand()%mxM+1 << endl;
    }
    //TODO: lehet, h. Európának még nincs vége!
    for(int i=eu+1;i<=n-am;++i) {
      if (rand()/(RAND_MAX+1.0)<pT)
      {
        oF << rand()%mxM+1 << endl;
      }
       else
      {
        oF << 0 << endl;
      }
    }
    //TODO: lehet, h. Amerika már elkezdődött!
    for(int i=n-am+1;i<=n;++i) {
      oF << rand()%mxM+1 << endl;
    }
    oF.close();
}
```

Véletlen tesztek
(Példa – C++)

```
//Fájlba generálás:
void fajlbaGeneralas(string fN, int n, int nEu, int nAm, float pT, int mxM)
//Uf: N sorban, soronként egyetlen 0..mxM közötti egész szám ...
{
    ofstream oF(fN.c_str());
    int eu=rand()%nEu+1;//Európa hossza
    int am=rand()%nAm+1;//Amerika hossza
    for(int i=1;i<=eu;++i) {
      oF << rand()%mxM+1 << endl;
    }
    //TODO: lehet, h. Európának még nincs vége!
    for(int i=eu+1;i<=n-am;++i) {
      if (rand()/(RAND_MAX+1.0)<pT)
      {
        oF << rand()%mxM+1 << endl;
      }
       else
      {
        oF << 0 << endl;
      }
    }
    //TODO: lehet, h. Amerika már elkezdõdött!
    for(int i=n-am+1;i<=n;++i) {
      oF << rand()%mxM+1 << endl;
    }
    oF.close();
```

Véletlen tesztek
(Példa – C++)



Alprogramok implementációja

Kód:

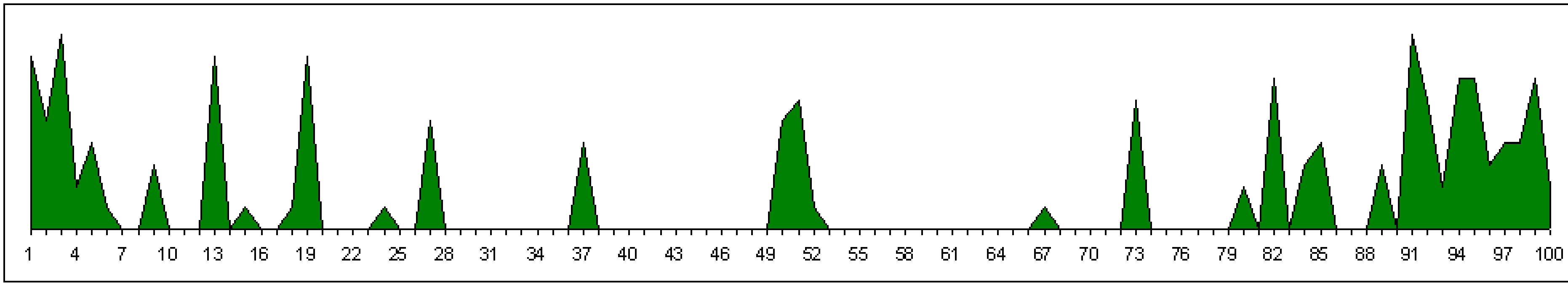
```
//név: Gipsz Jakab
//beolvassa a mnN..mxN közötti egész számot (mxN<mnN => végtelen)
//Fájlba generálás:
```

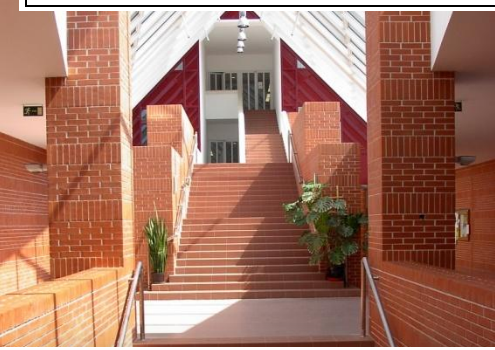
```
//beolvassa a mnN..mxN közötti egész számot (mxN<mnN => végtelen)
void be_int(string kerd, int &n, int mnN, int mxN, string uz)
{
    …
}

//beolvassa a mnN..mxN közötti valós számot (mxN<mnN => végtelen)
void be_float(string kerd, float &x, float mnN, int mxN, string uz)
{
    …
}

void billentyureVar()
{
    …
}
```

Kód jegyzet-ként

Véletlen tesztek
(Példa – C++)

Az eredményfájl





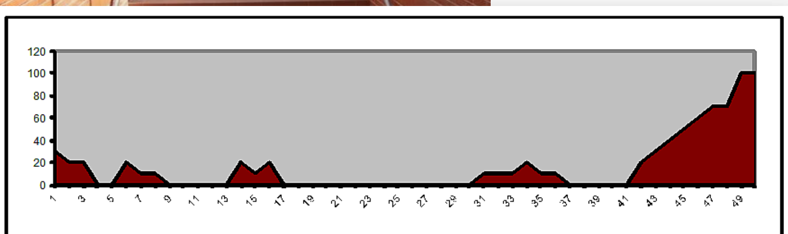
… és elemzése:

Bemeneti paraméterek

N =	100
NEu =	9
NAm =	9
MaxMag =	9
PSzig =	0,20

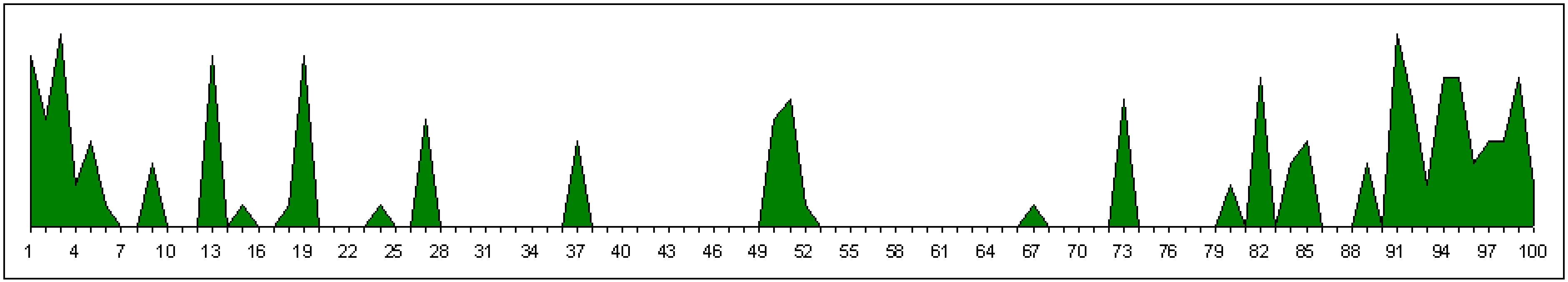
Kimeneti statisztika

P(Mag[i]>0) =	0,21
M(Mag[i]) =	1,49
M(Mag[i] | Mag[i]>0) =	3,83

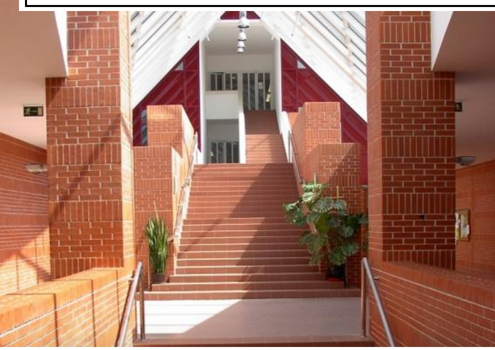


Véletlen tesztek
(Példa – C++)

Az eredményfájl



… és elemzése:

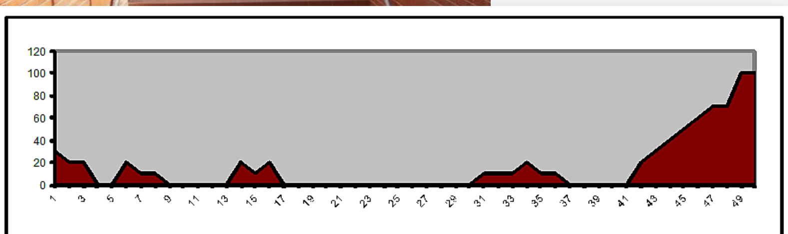


Bemeneti paraméterek

N =	100
NEu =	9
NAm =	9
MaxMag =	9
PSzig =	0,20

Kimeneti statisztika

P(Mag[i]>0) =	0,21
M(Mag[i]) =	1,49
M(Mag[i] | Mag[i]>0) =	3,83



Adatfájl jegyzet-ként

Programozási alapismeretek
10. előadás vége