OEP 1. gyakorlat

intervallumos programozási tételek
a megoldás előállítása szigorú visszavezetéssel

1. feladat

- Adottak az x és y nem negatív egész számok. Számítsuk ki a szorzatukat, úgy, hogy csak összedást használhatunk.
- Feladat állapottere (bemeneti és kimeneti változók és típusaik)
- A=(x:ℕ, y:ℕ, z:ℕ)
- Előfeltétel (mely változók rendelkeznek kezdőértékkel, valamint a kezdőértékre vonakozó elvárások)
- Ef=($x=x_0 \wedge y=y_0$) másképp írva: Ef=(x=x’ ∧ y=y’)
- Utófeltétel (utalunk a programozási tételre is)
- Uf=($z = \sum_{i=1}^{x_0} y_0$) vagy Uf=(Ef $\wedge\ z = \sum_{i=1}^{x} y$)

Összegzés programozási tétel

1. Összegzés

Feladat: Adott egy f:$[m..n] \rightarrow H$ függvény. A H halmaz elemein értelmezett egy asszociatív, baloldali nulla elemmel rendelkező művelet (nevezzük ezt összeadásnak és jelölje a +). Határozzuk meg a függvény intervallumon felvett értékeinek összegét!

Specifikáció:

$A = (m{:}\mathbb{Z},\ n{:}\mathbb{Z}\ ,\ s{:}H)$

$Ef = (\ m=m' \wedge n=n')$

$Uf = (\ Ef \wedge \mathrm{s} = \sum_{i=m..n} f(i))$

Algoritmus:

$s := 0$	
$i = m\ ..\ n$	$i{:}\mathbb{Z}$
$s := s+f(i)$	

Visszavezetés az összegzés tételre

- A=(x:ℕ, y:ℕ, z:ℕ)
- Ef=($x=x_0 \wedge y=y_0$)
- Uf=(Ef $\wedge\ z = \sum_{i=1}^{x} y$)

1. Összegzés

Feladat: Adott egy $f{:}[m..n]\to H$ függvény. A H halmaz elemein értelmezett egy asszociatív, baloldali nulla elemmel rendelkező művelet (nevezzük ezt összeadásnak és jelölje a +). Határozzuk meg a függvény intervallumon felvett értékeinek összegét!

Specifikáció:

$A = (m{:}\mathbb{Z},\ n{:}\mathbb{Z},\ s{:}H)$
$Ef = (\ m=m' \wedge n=n'\ )$
$Uf = (\ Ef \wedge s = \sum_{i=m..n} f(i))$

Algoritmus:

$s := 0$	
$i = m\ ..\ n$	$i{:}\mathbb{Z}$
$\quad s := s+f(i)$	

i=m..n	~	i=1..x
s	~	z
f(i)	~	y
H,+,0	~	N,+,0

Visszavezetéssel kapott megoldó algoritmus

- $A=(x:\mathbb{N},\ y:\mathbb{N},\ z:\mathbb{N})$
- $Ef=(x=x_0 \wedge y=y_0)$
- $Uf=(Ef \wedge z = \sum_{i=1}^{x} y\ )$

i=m..n	~	i=1..x
s	~	z
f(i)	~	y
H,+,0	~	N,+,0

1. Összegzés

Feladat: Adott egy f:[m..n]→H függvény. A H halmaz elemein értelmezett egy asszociatív, baloldali nulla elemmel rendelkező művelet (nevezzük ezt összeadásnak és jelölje a +). Határozzuk meg a függvény intervallumon felvett értékeinek összegét!

Specifikáció:

$A = (m:\mathbb{Z},\ n:\mathbb{Z},\ s:H)$

$Ef = (\ m=m' \wedge n=n')$

$Uf = (\ Ef \wedge s = \sum_{i=m..n} f(i))$

Algoritmus:

$s := 0$	
$i = m .. n$	$i:\mathbb{Z}$
$s := s+f(i)$	

z := 0	
i = 1..x	i:ℤ
z := z + y	

2. feladat

- Adott n darab valós szám, adjuk meg a legnagyobb abszolút értékű számot.
- Feladat állapottere (bemeneti és kimeneti változók és típusaik)
- A=(n:$\mathbb{N}$, x:$\mathbb{R}^n$, ind:$\mathbb{N}$,ertek:$\mathbb{R}$)
- Előfeltétel (mely változók rendelkeznek kezdőértékkel, valamint a kezdőértékre vonakozó elvárások)
- Ef=($n=n_0 \wedge x=x_0 \wedge n>0$)   másképp írva: Ef=(x=x' $\wedge$ n>0) //n=n' ebben benne van
- Utófeltétel (utalunk a programozási tételre is)
- Uf=(Ef $\wedge$ ertek,ind=$\mathrm{Max}_{i=1..n}|x[i]|$)

Maximum kiválasztás programozási tétel

3. Maximum kiválasztás

Feladat: Adott egy f:[$m..n$]→H függvény. A H halmaz elemein értelmezett egy teljes rendezési reláció. Határozzuk meg, melyik a függvény legnagyobb értéke és adjuk meg az egyik olyan intervallumbeli elemet, ahol a függvény ezt az értéket felveszi!

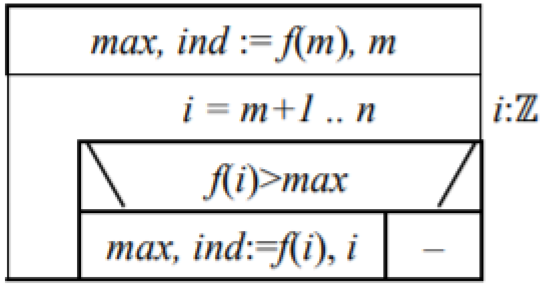
Specifikáció:

$A = (m:\mathbb{Z},\ n:\mathbb{Z},\ max:H,\ ind:\mathbb{Z})$

$Ef = (m=m' \wedge n=n' \wedge m \leq n)$

$Uf = (\ Ef \wedge\ max,ind = MAX_{i=m..n} f(i)\ )$

Algoritmus:



Visszavezetés maximum kiválasztás tételre

- $A=(n:\mathbb{N},\ x:\mathbb{R}^n,\ ind:\mathbb{N}, ertek:\mathbb{R})$
- $Ef=(n=n_0 \wedge x=x_0 \wedge n>0)$
- $Uf=(Ef \wedge ertek,ind=Max_{i=1..n}|x[i]|)$

3. Maximum kiválasztás

Feladat: Adott egy $f:[m..n]\to H$ függvény. A H halmaz elemein értelmezett egy teljes rendezési reláció. Határozzuk meg, melyik a függvény legnagyobb értéke és adjuk meg az egyik olyan intervallumbeli elemet, ahol a függvény ezt az értéket felveszi!

Specifikáció:

$A = (m:\mathbb{Z},\ n:\mathbb{Z},\ max:H,\ ind:\mathbb{Z})$

$Ef = (m=m' \wedge n=n' \wedge m\leq n)$

$Uf = (Ef \wedge max,ind = MAX_{i=m..n} f(i))$

Algoritmus:

$max, ind := f(m), m$

$i = m+1 .. n$ $i:\mathbb{Z}$

$f(i)>max$

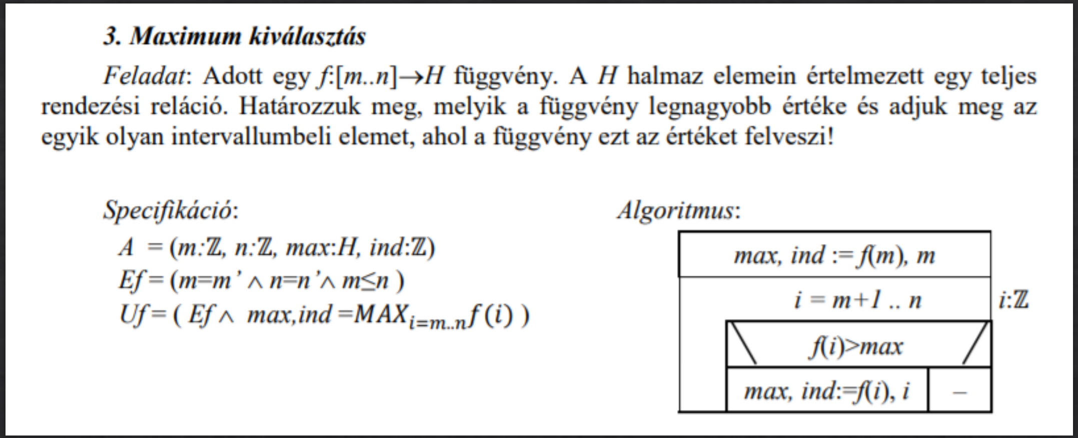
$max, ind := f(i), i$ | –

i=m..n	~	i=1..n
max,ind	~	ertek,ind
f(i)	~	|x[i]|
H,>	~	R,>

Visszavezetéssel kapott megoldó algoritmus

- $A=(n:\mathbb{N},\ x:\mathbb{R}^n,\ ind:\mathbb{N}, ertek:\mathbb{R})$
- $Ef=(n=n_0 \wedge x=x_0 \wedge n>0)$
- $Uf=(Ef \wedge ertek,ind=Max_{i=1..n}|x[i]|)$

m..n	~	1..n
max,ind	~	ertek,ind
f(i)	~	|x[i]|
H,>	~	R,>



3. *Maximum kiválasztás*

Feladat: Adott egy f:[$m..n$]→H függvény. A H halmaz elemein értelmezett egy teljes rendezési reláció. Határozzuk meg, melyik a függvény legnagyobb értéke és adjuk meg az egyik olyan intervallumbeli elemet, ahol a függvény ezt az értéket felveszi!

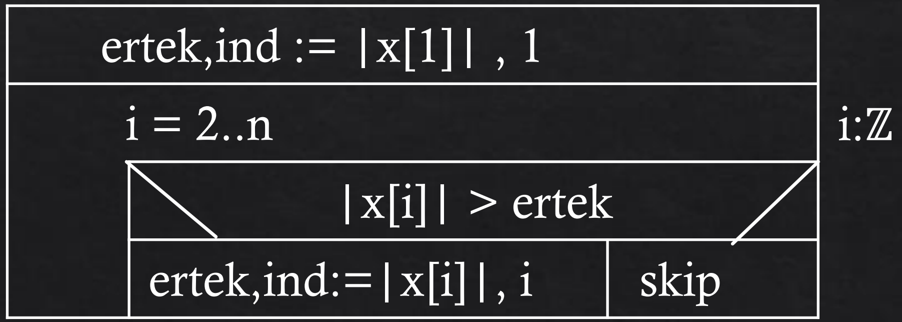
Specifikáció:

$A = (m:\mathbb{Z},\ n:\mathbb{Z},\ max:H,\ ind:\mathbb{Z})$

$Ef = (m=m' \wedge n=n' \wedge m \leq n)$

$Uf = (Ef \wedge max,ind = MAX_{i=m..n} f(i))$

Algoritmus:



3. feladat

- Adott n nap átlaghőmérséklete, keressük meg azt a napot, amikor hidegebb lett, mint az első napon volt.
- Feladat állapottere (bemeneti és kimeneti változók és típusaik)
- $A=(n:\mathbb{N},\ h:\mathbb{R}^n,\ van:\ \mathbb{L},\ ind:\mathbb{N})$
- Előfeltétel (mely változók rendelkeznek kezdőértékkel, valamint a kezdőértékre vonakozó elvárások)
- $Ef=(n=n_0 \land h=h_0)$
- Utófeltétel (utalunk a programozási tételre is)
- $Uf=(Ef \land van,ind=Search_{i=2..n}h[i]<h[1])$

Lineáris keresés programozási tétel

6. Keresés (szekvenciális vagy lineáris keresés)

Feladat: Adott egy *felt*:[*m*..*n*]→$\mathbb{L}$ feltétel. Határozzuk meg az intervallum első olyan elemét, amelyre teljesül a feltétel! (Ez az ún. pesszimista lineáris keresés.)

Specifikáció:

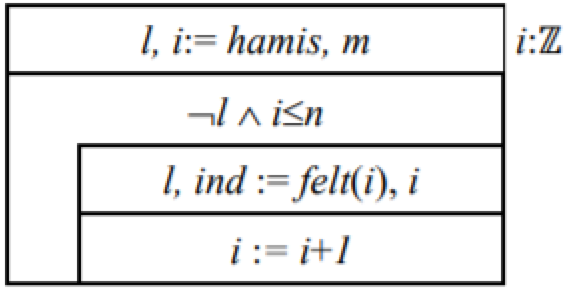
$A = (m:\mathbb{Z},\ n:\mathbb{Z},\ l:\mathbb{L},\ ind:\mathbb{Z})$

$Ef = (m=m' \wedge n=n')$

$Uf = (Ef \wedge l, ind =$

$SEARCH_{i=m..n} felt(i))$

Algoritmus:



Visszavezetés lineáris keresés tételre

- $A=(n:\mathbb{N},\ h:\mathbb{R}^n,\ van:\ \mathbb{L},\ ind:\mathbb{N})$
- $Ef=(n=n_0 \wedge h=h_0)$
- $Uf=(Ef \wedge van,ind=Search_{i=2..n} h[i]<h[1])$

6. *Keresés (szekvenciális vagy lineáris keresés)*

Feladat: Adott egy *felt*:[*m*..*n*]→𝕃 feltétel. Határozzuk meg az intervallum első olyan elemét, amelyre teljesül a feltétel! (Ez az ún. pesszimista lineáris keresés.)

Specifikáció:

$A = (m:\mathbb{Z},\ n:\mathbb{Z},\ l:\mathbb{L},\ ind:\mathbb{Z})$

$Ef = (m=m' \wedge n=n')$

$Uf = (Ef \wedge l, ind = SEARCH_{i=m..n} felt(i))$

Algoritmus:

l, i:= hamis, m	*i*:ℤ
$\neg l \wedge i \leq n$	
l, ind := felt(i), i	
i := i+1	

m..n	~	2..n
l,ind	~	van,ind
felt(i)	~	h[i]<h[1]

Visszavezetéssel kapott megoldó algoritmus

- $A=(n:\mathbb{N},\ h:\mathbb{R}^n,\ van:\ \mathbb{L},\ ind:\mathbb{N})$
- $Ef=(n=n_0 \wedge h=h_0)$
- $Uf=(Ef \wedge van,ind=Search_{i=2..n}h[i]<h[1])$

m..n	~	2..n
l,ind	~	van,ind
felt(i)	~	h[i]<h[1]

6. *Keresés (szekvenciális vagy lineáris keresés)*

Feladat: Adott egy *felt*:[*m..n*]→$\mathbb{L}$ feltétel. Határozzuk meg az intervallum első olyan elemét, amelyre teljesül a feltétel! (Ez az ún. pesszimista lineáris keresés.)

Specifikáció:

$A = (m:\mathbb{Z},\ n:\mathbb{Z},\ l:\mathbb{L},\ ind:\mathbb{Z})$
$Ef = (m=m' \wedge n=n')$
$Uf = (Ef \wedge l,\ ind = SEARCH_{i=m..n} felt(i))$

Algoritmus:

l, i:= hamis, m		*i*:$\mathbb{Z}$
$\neg l \wedge i \le n$		
	l, ind := felt(i), i	
	i := i+1	

van, i:= hamis, 2		i:$\mathbb{Z}$
$\neg$ van $\wedge$ i $\le$ n		
	van, ind:= h[i] < h[1], i	
	i:=i+1	

4. feladat

- Adott egy egészeket tartalmazó n méretű tömb. Melyik érték fordul elő benne legtöbbször?
- Feladat állapottere
- A=(n:ℕ, x:$\mathbb{Z}^n$, ind:ℕ,max:ℤ)
- Előfeltétel
- Ef=(x=x' ∧ n'>0)
- Utófeltétel
- Uf=(Ef ∧ max,ind=$\mathrm{Max}_{i=1..n}$ hanyszor(i) és

  hanyszor(i)=$\sum_{\substack{j=1 \\ x[i]=x[j]}}^{n} 1$)

Visszavezetés

- Külső tétel: maximum kiválasztás

3. Maximum kiválasztás

Feladat: Adott egy f:[$m..n$]$\to H$ függvény. A H halmaz elemein értelmezett egy teljes rendezési reláció. Határozzuk meg, melyik a függvény legnagyobb értéke és adjuk meg az egyik olyan intervallumbeli elemet, ahol a függvény ezt az értéket felveszi!

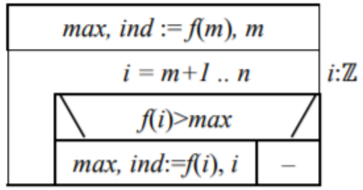
Specifikáció:

$A = (m:\mathbb{Z},\ n:\mathbb{Z},\ max:H,\ ind:\mathbb{Z})$

$Ef = (m=m' \wedge n=n' \wedge m \leq n)$

$Uf = (\ Ef \wedge\ max,ind = MAX_{i=m..n} f(i)\ )$

Algoritmus:



i=m..n	~	i=1..n
max,ind	~	max,ind
f(i)	~	hanyszor(i)
H,>	~	N,>

- Belső tétel: számlálás

2. Számlálás

Feladat: Adott egy $felt$:[$m..n$]$\to\mathbb{L}$ feltétel. Határozzuk meg, hogy hányszor teljesül az intervallumon a feltétel, azaz hányszor veszi fel az igaz értéket!

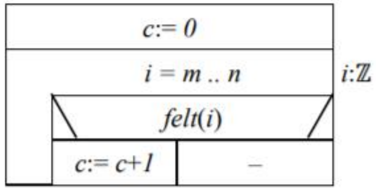
Specifikáció:

$A = (m:\mathbb{Z},\ n:\mathbb{Z},\ c:\mathbb{N})$

$Ef = (\ m=m' \wedge n=n'\ )$

$Uf = (\ Ef \wedge c = \sum_{\substack{i=m..n \\ felt(i)}} 1\ )$

Algoritmus:

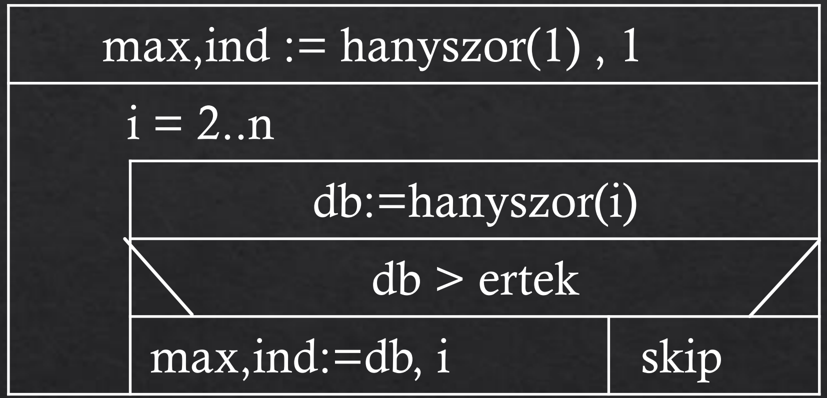


i	~	j
i=m..n	~	j=1..n
c	~	c
felt(i)	~	x[i]=x[j]

Visszavezetéssel kapott algoritmusok

- Maximum kiválasztás

i=m..n	~	i=1..n
max,ind	~	max,ind
f(i)	~	hanyszor(i)
H,>	~	N,>

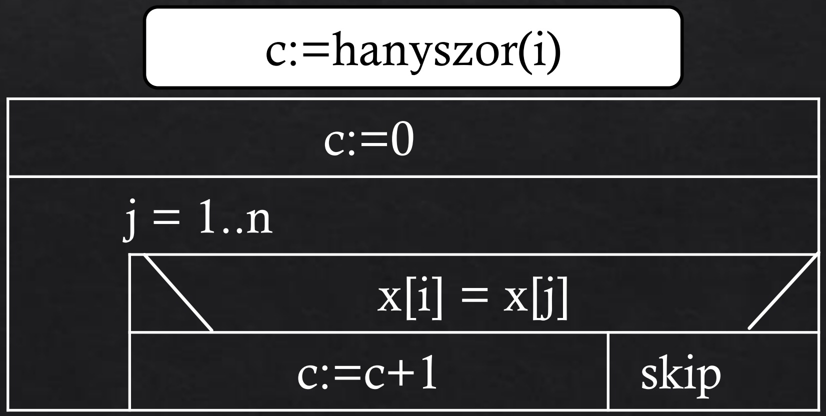


i:ℤ

db:ℤ

- Számlálás

i	~	j
i=m..n	~	j=1..n
c	~	c
felt(i)	~	x[i]=x[j]



j:ℤ