I. Hányadik órában volt a múzeumban a legtöbb látogató és mennyi?

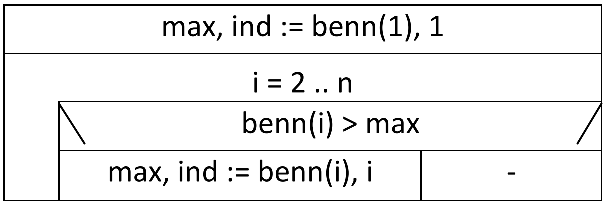
$A = (be : \mathbb{N}^n,\ ki : \mathbb{N}^n,\ max : \mathbb{N},\ ind : \mathbb{N})$

$Ef = (be=be' \wedge ki=ki' \wedge n>0)$

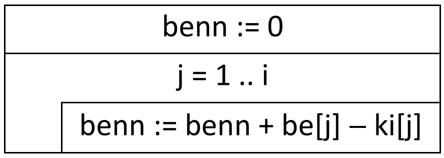
$Uf = (Ef \wedge (max, ind) = \mathrm{MAX}_{i=1..n}\ benn(i))$

ahol $benn : [1 .. n] \longrightarrow \mathbb{N}$ és

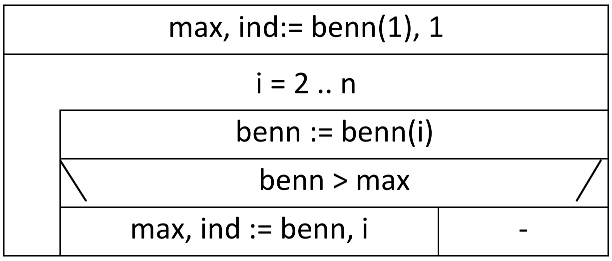
$benn(i) = \Sigma_{j=1..i}\ (be[j] - ki[j])$



benn := benn(i)



sőt

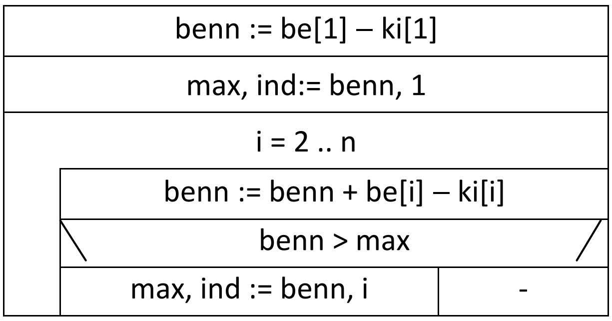


Másképpen

$benn(0) = 0$

$benn(i) = benn(i-1)+be[i] - ki[i]$ (ha $i \geqq 1$)

A kezdeti *max,ind:=benn(1),1* értékadásból kiemeljük a *benn(1)*-t a *benn:=benn(1)* értékadásba, ahol a *benn(1)* a *be[1]–ki[1]* alakban is megadható. Ezután „kibontjuk” a rekurzív függvényt, azaz a ciklusmagbeli *benn:=benn(i)* értékadást a *benn:=benn+be[i]–ki[i]* értékadással helyettesítjük. Erre az ad lehetőséget, hogy egyrészt a rekurzív definíció miatt a *benn(i)* azonos a *benn(i–1)+be[i]–ki[i]* kifejezéssel, másrészt a ciklusmag elején feltételezhetjük a *benn=benn(i–1)* összefüggést.



II. Keressük meg az első 5 kilométernél hosszabb sziget kezdetét!

$A = (x : \mathbb{R}^n,\ l : \mathbb{L},\ kezdet : \mathbb{N})$

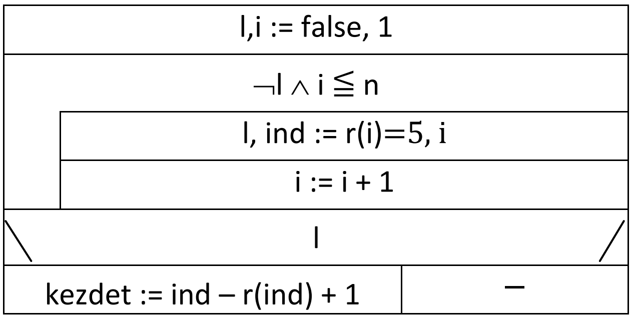
$Ef = (\ x = x'\ )$

ahol $r : [0\,..\,n] \longrightarrow \mathbb{N}$ és

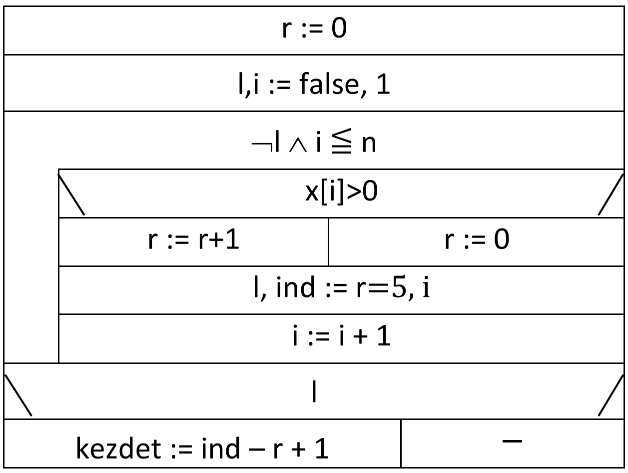
$$r(0) = 0$$

$$r(i) = \begin{cases} r(i-1)+1 & \text{ha } x[i]>0 \\ 0 & \text{ha } x[i]=0 \end{cases} \qquad (\ i \geqq 1)$$

$Uf = (\ Ef\ \wedge (l, ind) = \mathrm{SEARCH}_{i=1..n}\ (r(i) = 5)\ \wedge (l \longrightarrow kezdet = ind - r(ind) + 1)\ \ )$



Kiemelve az r segédváltozóba a rekurzív függvényt, majd kibontva azt, megkapjuk a végleges programot:



r := r(0)

r := r(i)

III. Kódoljunk egy adott szöveget (text = „zh”) – vagy annak egy részét – egy adott rejtjelező kulcs (key = József Attila: Altató) alapján egy indexsorozattal!

$A = (key : \mathbb{K}^n,\ text : \mathbb{K}^m,\ code : \mathbb{N}^*)$

$Ef = (\ key=key' \wedge text=text')$

$$Uf = \Big(\ Ef \ \wedge\ code = \bigoplus_{\substack{i=1..n \\ last(i-1)<m \ \wedge\ key[i] = text[last(i-1)+1]}} \langle i \rangle\ \Big)$$

ahol $last(i-1)$ ~ a text-nek hány karakterét lehet a key első $i-1$ karakterének segítségével kódolni; azaz a text soron következő kódolandó karaktere a $last(i-1)+1$-dik.

Tehát $last : [0\,..\,n] \longrightarrow \mathbb{N}$ és

$$last(0) = 0$$

$$last(i) = \begin{cases} last(i-1)+1 & \text{ha } last(i-1)<m \wedge key[i]=text[last(i-1)+1] \\ last(i-1) & \text{különben} \end{cases} \quad (\,i \geqq 1)$$

```
code := < >
i = 1 .. n
    last(i–1)<m ∧ key[i] = text[last(i–1)+1]
        code := code ⊕ <i>   |   –
```

Kiemeljük a rekurzív függvényt a last segédváltozóba, de ügyelni kell arra, hogy mivel a ciklusmagban a last(i–1)-re hivatkozunk, ezért last(i)-t a ciklusmag végén kell a segédváltozónak értékül adni:

```
code, last := < >, last(0)
i = 1 .. n
    last<m ∧ key[i] = text[last+1]
        code := code ⊕ <i>   |   –
    last := last(i)
```

Kibontva a rekurzív függvényt és összevonva a ciklusmagbeli két elágazást:

```
code, last := < >, 0
i = 1 .. n
    last<m ∧ key[i] = text[last+1]
        last, code := last+1, code ⊕ <i>   |   –
```

last<m ∧

Megjegyzés:

1. A last tulajdonképpen a code hosszát mutatja.
2. Leálláskor a last<m feltétel azt mutatja, ha nem sikerült teljesen lekódolni a szöveget.
3. Javítható (gyorsítható) a megoldás, ha a ciklus feltételbe belevesszük a last<m feltételt is, hiszen last≧m esetén már elkészült a kód. Ekkor a ciklusbeli elágazás feltételében feleslegessé válik a last<m ellenőrzés.

$$Uf = \left(Ef \wedge code = \bigoplus_{\substack{i=1..n \\ key[i] = text[last(i-1)+1]}}^{last(i-1) < m} \langle i \rangle \right)$$

(Jól jönne tehát egy ilyen feltételig tartó összegzés.)

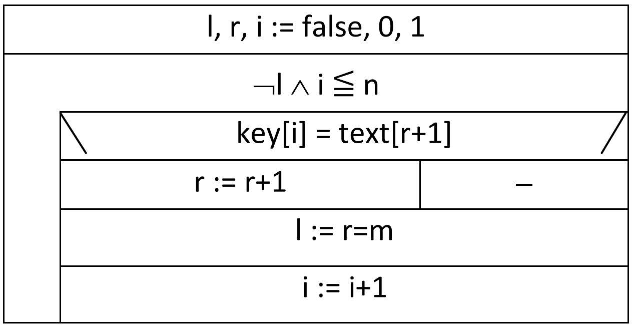
A feladat korábbi verziója:

Lehet-e kódolni egy adott szöveget egy adott rejtjelező kulcs (szöveg) alapján egy indexsorozattal?

A = (key : $\mathbb{K}^n$, text : $\mathbb{K}^m$, l : $\mathbb{L}$)

Ef = (key=key' $\wedge$ text=text')

Uf = (Ef $\wedge$ l = $\exists_{i=1..n}$ last(i)=m)



IV. Melyik a legnagyobb összegű szakasz?

A = (x : $\mathbb{R}^n$, kezdet : $\mathbb{N}$, vég : $\mathbb{N}$, max : $\mathbb{R}$)

Ef = (x=x' $\wedge$ n$>$0)

Uf = (Ef $\wedge$ ((max, kezdet), vég) = $\mathrm{MAX}_{i=1..n}$ r(i))

ahol az r(i) az x tömb i-dik eleménél végződő szakaszai közül a legnagyobb összegűnek az összegét és kezdő elemének indexét adja meg, azaz

r : [1 .. n] $\longrightarrow \mathbb{R}\times\mathbb{N}$

r(i) = $\mathrm{MAX}_{j=1..i}(\boldsymbol{\Sigma}_{k=j..i}\ x[k])$)

A külső maximum kiválasztás az r(i) párok között keresi meg azt, amelyiknek az összege a legnagyobb. Ez tehát összeg-kezdet számpárokat hasonlít össze nyilvánvalóan a számpárok első komponense, az összeg alapján. Az eredmény egyrészt a maximális érték, amely most a (max, kezdet) számpár lesz, másrészt a az index, amely most a vég.

Az r függvény rekurzívan ($r(i)_1$ az r(i) első komponensét, $r(i)_2$ az r(i) második komponensét jelöli):

r(1) = (x[1], 1)

r(i) = (x[i], i) ha $x[i]>r(i-1)_1+x[i]$ (i≧2)

$(r(i-1)_1+x[i],\ r(i-1)_2)$ kül

(max, kezdet), vég:= r(1), 1		
i = 2 .. n		
	$r(i)_1$ > max	
	(max, kezdet), vég := r(i), i	–

Kiemelve, majd kibontva a rekurzív függvényt az össz és k segédváltozókba:

össz, k := x[1] ,1			(össz, k) := r(1)
max, kezdet, vég:= össz, k, 1			
i = 2 .. n			
	x[i] > össz+x[i]		
	össz, k := x[i], i	össz := össz +x[i]	(össz, k) := r(i)
	össz > max		
	max, kezdet, vég := össz, k, i	–	