

## 5. Óra

### Miről lesz szó?

### Szöveges állomány adatainak felsorolása, *TextFile.dll*

<https://people.inf.elte.hu/gt/oep/TextFile.zip>

#### Használat

1. Másoljuk bele a projektünk mappájába (nem feltétlen szükséges, csak később ne mozgassuk, ha beállítottuk a referenciát)
2. Állítsunk referenciát a dll-re (**Dynamic Link Library**)
  1. Jobb klikk Dependencies fülre a projekt alatt
  2. Add Project Reference
  3. Felugró ablakban: Browse...
  4. Keressük meg a .dll file-t és adjuk hozzá

Text File Reader instance:

```
using TextFile;  
  
TextFileReader reader = new("<input txt file path>");
```

Publikus metódusai:

```
public bool ReadChar(out char ch);  
public char? ReadChar();  
  
public bool ReadSign(out char ch); // Átugorja a whitespace-eket: ' ' \n \t \r  
public char? ReadSign();          //  
  
public bool ReadString(out string str);  
public string ReadString(); // Ha nincs mit olvasni: ""-el tér vissza  
  
public bool ReadInt(out int n);  
public int? ReadInt();  
  
public bool ReadDouble(out double a);  
public double? ReadDouble();  
  
public bool ReadLine(out string line);  
public string ReadLine(); // ez is lehet null!
```

```
TextFileReader x = new TextFileReader("<input txt file path>");  
  
st, e, x : read ⇔ st = x.ReadEType(out e); // x: infile (input file)  
st = norm      ⇔ st = true
```

# Feladatok

## Kaktuszosdi

- Válogassuk ki kaktuszok sorozatából egyrészt a piros virágú kaktuszoknak, másrészt a mexikói őshazájú kaktuszoknak neveit!

Specifikáció:

$A = (x : \text{enor}(\text{Kaktusz}), y, z : \mathbb{S}^*)$

$\text{Kaktusz} = \text{rec}(\text{név}:\mathbb{S}, \text{szín}:\mathbb{S}, \text{ős}:\mathbb{S}, \text{méret}:\mathbb{N})$

$Ef = (x = x_0)$

$Uf = (y = \bigoplus_{e \in x_0} \langle e.\text{név} \rangle \wedge z = \bigoplus_{e \in x_0} \langle e.\text{név} \rangle)$   
 $\quad \quad \quad e.\text{szín} = \text{"piros"} \quad \quad \quad e.\text{ős} = \text{"Mexikó"}$

2 összegzés (kiválogatás) közös ciklusba vonva

$H, +, 0 \sim (\mathbb{S}^*, \oplus, <>), (\mathbb{S}^*, \oplus, <>)$

$f_1(e) \sim \langle e.\text{név} \rangle$  ha  $e.\text{szín} = \text{"piros"}$

$f_2(e) \sim \langle e.\text{név} \rangle$  ha  $e.\text{ős} = \text{"Mexikó"}$

Szekvenciális inputfájltra:

$y, z := <>, <>$ $st, e, x : \text{read}$	$e : \text{Kaktusz}$ $st : \text{Status}$
$st = \text{norm}$	
$e.\text{szín} = \text{"piros"}$	
$y := y \oplus \langle e.\text{név} \rangle$	—
$e.\text{ős} = \text{"Mexikó"}$	
$z := z \oplus \langle e.\text{név} \rangle$	—
$st, e, x : \text{read}$	

Algoritmus:

$y, z := <>, <>$	
$e \text{ in } x$	$e : \text{Kaktusz}$
$e.\text{szín} = \text{"piros"}$	
$y := y \oplus \langle e.\text{név} \rangle$	—
$e.\text{ős} = \text{"Mexikó"}$	
$z := z \oplus \langle e.\text{név} \rangle$	—

Lehetne indexeléssel is implementálni, és ehhez akár számlálós ciklust ( $i = 1 \dots |x|$ ) is használhatunk.

Tömbre:

$y, z := <>, <>$ $i := 1$	$i : \mathbb{N}$
$i \leq  x $	
$x[i].\text{szín} = \text{"piros"}$	
$y := y \oplus \langle e.\text{név} \rangle$	—
$x[i].\text{ős} = \text{"Mexikó"}$	
$z := z \oplus \langle e.\text{név} \rangle$	—
$i := i + 1$	

Szimultán maximum kiválasztás és optimista keresés összegzésre visszavezetve

2. Keressük meg egy pozitív egész számokat tartalmazó nem üres sorozatban a legnagyobb számot, és közben döntsük el azt is, hogy vajon van-e páros szám.

*Specifikáció:*

$$A = (x : \text{enor}(\mathbb{N}^+), l : \mathbb{L}, m : \mathbb{N})$$

$$Ef = (x = x_0 \wedge |x| \geq 1)$$

$$Uf = ((m, \_) = \mathbf{MAX}_{e \in x_0} e \wedge (l, \_) = \mathbf{SEARCH}_{e \in x_0} (e \text{ páros}))$$

*Az utófeltétel másképpen: két összegzéssel*

$$Uf = (m = \mathbf{MAX}_{e \in x_0} e \wedge l = \mathbf{V}_{e \in x_0} (e \text{ páros}))$$

$$\begin{array}{ll} \text{ahol } \underline{\max}: \mathbb{N} \times \mathbb{N} \rightarrow \mathbb{N}, \underline{\max}(a, b) ::= \max(a, b) & \text{neutrális elem: } 0 \\ \vee: \mathbb{L} \times \mathbb{L} \rightarrow \mathbb{L}, \quad \vee(a, b) ::= a \vee b & \text{neutrális elem: igaz} \end{array}$$

*Két összegzés összevonva*

$$H, +, 0 \sim (\mathbb{N}, \underline{\max}, 0), (\mathbb{L}, \vee, \text{hamis})$$

$$f(e) \sim e, e \text{ páros}$$

$$s \sim m, l$$

*Algoritmus:*

m, l := 0, hamis		
e in x		e : ℕ
e > m		
m := e	—	m := max(m,e)
l := l ∨ e páros		

**Páros számok darabszáma az első negatív szám előtt, és azt követően azzal együtt**

d) Hány páros szám van az első negatív számot megelőzően, és hány azt követően azzal együtt?

Specifikáció:

$A = (x:infile(\mathbb{Z}), dbe, dbu:\mathbb{N})$

$Ef = (x=x_0)$

$Uf = (dbe, e', x') = \sum_{\substack{e \in x_0 \\ e \text{ páros}}}^{e \geq 0} 1$

$\wedge dbu = \sum_{\substack{e \in \langle e' \rangle \oplus x' \\ e \text{ páros}}} 1$

Az  $e'$  az  $e$  változó értéke,  $x'$  az  $x$  felsoroló állapota az első számlálás leállásakor.

A második számlálás úgy folytatja  $x$  felsorolását, hogy előtte figyelembe veszi az  $e'$  elemet is.

Algoritmus:

dbe := 0		e:ℤ
e in x ∧ e ≥ 0		
e páros		
dbe := dbe + 1	—	
dbu := 0		
e in <e, x>		
e páros		
dbu := dbu + 1	—	

Szekvenciális inputfájltra:

dbe := 0		e:ℤ st : Status
st, e, x : read		
st=norm ∧ e≥0		
e páros		
dbe := dbe +1	—	
st, e, x : read		
dbu := 0		
st=norm		
e páros		
dbu := dbu+1	—	
st, e, x : read		

Tömbre:

dbe := 0		e:ℤ i:ℕ
i:=1		
i≤n ∧ x[i]≥0		
x[i] páros		
dbe := dbe + 1	—	
i := i+1		
dbu := 0		
i≤n		
x[i] páros		
dbu := dbu+1	—	
i := i+1		

**Kisbeadandó (5/10) (3. batch)**

4. Egymás utáni napok átlaghőmérsékleteit egy szekvenciális inputfájl tartalmazza. Mennyi az első fagypont alatti értéket megelőző napok (ilyenek biztosan vannak) hőmérsékleteinek átlaga, továbbá az első fagypont alatti értéktől kezdődően (az első fagypont alatti napot is beleértve) vajon minden nap fagypont alatt maradt-e a hőmérséklet, és mi volt a legalacsonyabb hőmérséklet?

*Specifikáció:*

$A = (x:infile(\mathbb{R}), a:\mathbb{R}, l:\mathbb{L}, kicsi:\mathbb{R})$

$Ef = (x=x_0 \wedge \exists i \in [2..|x|]: x[i] < 0 \wedge x[1] \geq 0)$

$Uf = ((s, e', x') = \sum_{e \in x_0}^{e \geq 0} (e) \wedge (db, e', x') = \sum_{e \in x_0}^{e \geq 0} 1 \wedge a = s/db \wedge$   
 $\wedge l = \text{VSEARCH}_{e \in \langle e' \rangle \oplus x'} (e < 0) \wedge kicsi = \text{MIN}_{e \in \langle e' \rangle \oplus x'} e)$

*Algoritmus:*

$e, s:\mathbb{R}, db:\mathbb{N} \text{ st:Status}$

két összegzés közös ciklusban  
szekvenciális inputfájl feltételig tartó  
felsorolásával

átlagszámítás

*opt. lin. ker. és min. kiv.* közös ciklusban  
inputfájl felsorolásának folytatásával,  
ahol az első elemet már korábban beolvastuk,  
ahol az  $(l, kicsi := e < 0, e)$  inicializálás helyett,  
mivel  $e$  biztosan negatív szám,  
írhatjuk, hogy  $l, kicsi := igaz, e$

s, db := 0.0, 0	
st, e, x : read	
st=norm $\wedge e \geq 0$	
s, db := s+e, db +1	
st, e, x : read	
a := s / db	
l, kicsi := igaz, e	
st, e, x : read	
st=norm	
l := l $\wedge e < 0$	
e < kicsi	
kicsi := e	—
st, e, x : read	