

MÉGJEGYZÉSEK AZ 1. ZH-HOZ

Az alábbi visszatérő hibákkal találkoztam a zh javítása során.

1. Nem tiszta még mindig mindenki előtt az algoritmikus nyelv 'amíg-os ciklus'-ának szemantikája. Vagyis, hogy addig kell a ciklusmagot végrehajtani, amíg a feltétel logikai értéke IGAZ, más szóval akkor kell a ciklust elhagyni, amikor a feltétel HAMIS. Nem csökkenti az elírást az a tény, hogy van olyan programozási nyelv, amelyben van olyan ciklusféle, amiben a kilépés feltételét kell megfogalmazni.
2. A 'számlálós' ciklus (mint a Pascal-beli 'For'-ciklus) ciklusváltozójának direkt módosítás tilos, mivel ezt a ciklust szervező kód automatikusan és definíciószerűen végzi. Ha ok van arra, hogy a ciklusváltozó értéke időnként „szabálytalanul” módosuljon, akkor 'amíg-os ciklus'-sal kell az algoritmust (és a kódot) megvalósítani.

<i>Helytelen</i>	<i>Helyes</i>
<pre> ... Változó i,N:Egész ... Ciklus i=1-től N-ig ... Ha f(i) akkor i:=N+1 [a Szerző ki akar lépni] Ciklus vége ... Ciklus i=1-től N-ig ... i:+2 [a Szerző kettesével szeretne haladni a feldolgozásban] Ciklus vége ... </pre>	<pre> ... Változó i,N:Egész ... i:=1 Ciklus amíg i≤N és nem f(i) ... Ciklus vége ... Ciklus i=1-től N-ig 2-esével ... Ciklus vége ... </pre>
	<i>Helyes Pascal kód</i>
	<pre> ... Var i,N:Integer; ... i:=1; While (i≤N) and not f(i) do Begin ... End; ... i:=1; While i≤N do Begin ... Inc(i,2); End; ... </pre>

3. Nem mindegy –sokszor– a több tényezős vagy tagos feltételek rész kifejezéseinek sorrendje!

<i>Helytelen</i>	<i>Helyes</i>
Ciklus amíg $T(x(i))$ és $i \leq N$	Ciklus amíg $i \leq N$ és $T(x(i))$
...	...
Ciklus vége	Ciklus vége

Magyarázat: mivel a logikai műveletek (többnyire¹) nem szimmetrikusak, azaz ha a feltétel logikai értéke már kiderült, akkor a továbbiak kiértékelése elhagyandó. Így a helyes példában $i > N$ -re a $T(x(i))$ nem értékelődik ki, azaz index-túllépési hiba nem történik, amint a helytelen esetében.

4. Az *eljárás* (függvény és operátor) *hívásakor*² felesleges, sőt tilos az aktuális paraméterek *típusának* explicit kiírása.

<i>Helytelen</i>	<i>Helyes</i>
...	...
Változó aktPar ₁ :Típus ₁ aktPar ₂ :Típus ₂	Változó aktPar ₁ :Típus ₁ aktPar ₂ :Típus ₂
...	...
Elj(aktPar ₁ : Típus₁ , aktPar ₂ : Típus₂)	Elj(aktPar ₁ , aktPar ₂)
...	...

Megjegyzem: a híváskor persze a *hozzáférési jog* (Változó/Konstans) megadása éppen úgy tilos, mint a típusé! Az persze természetes, hogy az eljárás (függvény és operátor) *definiálásakor* viszont szükségesek:

<i>Helytelen</i>	<i>Helyes</i>
...	...
Eljárás Elj(aktPar ₁ , aktPar ₂): ... eljárás törzse ...	Eljárás Elj(Konstans aktPar ₁ :Típus ₁ , Változó aktPar ₂ :Típus ₂): ... eljárás törzse ...
Eljárás vége.	Eljárás vége.

5. Sokak számára az *utófeltétel célja* (s nem „csupán” a formalizmusa) homályos. Az utófeltétel **NEM** az **ALGORITMUS** leírás szavakkal! Célja, hogy precízen (ezért leginkább formális eszközökkel) fogalmazzuk meg a program (részprogram=eljárás/függvény/operátor) által elérendő célt (=adatváltozást), nem a célhoz vezető utat.

¹ Ha mást nem mondunk. L. a '[Programtranszformációk](#)'-ről szóló (2001.11.12-i) előadást.

² Remélem, itt és most definiálás nélkül is világos a 'hívás' szó jelentése

6. Az **operátor** olyan függvény vagy eljárás, amely hívási szintaxisa eltér a függvényekétől, vagy eljárásokétól. Vannak „hagyományos” bináris, infix jelöléssel használatosak (pl. „=”, vagy „:=”), de vannak „szokatlanabb” szintaxissal alkalmazandók is (pl. „Be:”, vagy tömb-indexelés).

Helytelen	Helyes
<p>Függvény Kisebb(Konstans s1, s2:Szöveg):Logikai</p> <p>...</p> <p>Kisebb:=...</p> <p>Függvény vége.</p> <p><i>ekkor az illeszkedő hívás alakja:</i></p> <p>...</p> <p>Ha Kisebb(s,ss) akkor ...</p>	<p>Infix Operátor Kisebb(Konstans s1, s2:Szöveg):Logikai</p> <p>Másként s1<s2 [itt derül ki a hívás- kori műveleti jel]</p> <p>...</p> <p>Kisebb:=...</p> <p>Operátor vége.</p> <p><i>ekkor az illeszkedő hívás alakja:</i></p> <p>...</p> <p>Ha s<ss akkor ...</p>

7. Sajnálatos módon sokan nem ismerik a Szöveg-típust. Közhiedelem, hogy karaktertömb, aminek értelmes dolog adottadik elemére hivatkozni. Ki kell jelentenem, a Szöveg olyan típus, amelyhez azon műveletgarnitúra egyike van hozzárendelve, amelyek az előadáson elhangzottak (vagy a mikrológiában, PowerPoint-os [bemutatón](#) olvashatók).

Nagy-nagy tisztelettel hívom föl a Hallgató figyelmét, hogy az 1-4. hibák bármelyikének elkövetésének egyes gyakorlati jegyet von maga után.