

Programozási nyelvek I. 5. gyakorlat

Balogh Ádám
bas@elte.hu

Eötvös Loránd Tudományegyetem
Informatikai Kar

Segítség az 1. feladathoz (1)

- Sablon specifikációja:
`generic`
Sablon paraméterek
Alprogram vagy csomag specifikációja
- (Korlátozott) átlátszatlan típus sablon paramétereként:
`type Név [(Diszkrimináns)] is`
`[limited] private;`
- Konstans sablon paramétereként:
`Név : in Típus [:= Érték];`

2004. október 14.

Programozási nyelvek I. - 5. gyakorlat

2

Segítség az 1. feladathoz (2)

- Sablon magában nem használható, előbb példányosítani kell
- A `with`-hez mindenképp oda kell írni ekkor is
- Sablon csomag példányosítása (fordítási egységen belül):
`package Példány is new`
`Sablon (Paraméterek);`
- Ezután már lehet `use Példány;`
- Sablon alprogram hasonlóan

2004. október 14.

Programozási nyelvek I. - 5. gyakorlat

3

1. feladat

Módosítsd a múlt órai diszkriminációs rekord segítségével megvalósított verem csomagot úgy, hogy az elemek típusa, a verem megadható legnagyobb mérete és a diszkriminációs alapértelmezett értéke a sablon paraméterei legyenek! Ezeknek a konstans sablon paramétereknek továbbra is 1000 illetve 100 legyen az alapértelmezett értéke. Módosítsd a kipróbáló programot is, hogy példányosítsa a sablont! A csomag neve legyen `Verem_DG`!

2004. október 14.

Programozási nyelvek I. - 5. gyakorlat

4

Segítség a 2. feladathoz

- Függvény, mint sablon paraméter:

```
with function Név ( Paraméterek )
    return Típus;
```
- Eljárás, mint sablon paraméter

```
with procedure Név ( Paraméterek ) ;
```
- Példányosításkor függvény paraméter helyére függvény jellegű attribútum is megadható (pl. `Típus`)

2004. október 14.

Programozási nyelvek I. - 5. gyakorlat

7

2. feladat

Írj egy `Kiír` sablon eljárást, ami egy tetszőleges átlátszatlan típust kiír. A sablon paramétere a típuson kívül legyen még az a függvény is, ami egy adott típusú értéket karakterlánccá alakít. A főprogram példányosítsa ezt a sablont egészekre!

2004. október 14.

Programozási nyelvek I. - 5. gyakorlat

8

Segítség a 3. feladathoz

- Diszkrét típus, mint sablon paraméter:
`type Név is (<>);`
- Tömb típus, mint sablon paraméter:
`type Név is array (Indexek)
of Típus;`
- Alprogramoknak, mint sablon paramétereknek is lehet feltételezett értéke, pl. a paraméter nevével megegyező nevű alprogram: a végére
`:= <>`

2004. október 14.

Programozási nyelvek I. - 5. gyakorlat

11

3. feladat

Írj sablon *Szumma* függvényt, amely egy vektor elemeit összegzi. A vektor indexeinek, elemeinek és magának a vektornak a típusa legyen a sablon paramétere, ugyanúgy, mint az összeadás művelete és a zéruselem. Az összeadás műveletének legyen alapértelmezett értéke a szokásos műveleti jel. Példáymosisd a sablont úgy, hogy produktumot számoljon!

2004. október 14.

Programozási nyelvek I. - 5. gyakorlat

12

Házi feladatok (1)

- Alakítsd át a sor típust is sablonná!
- Írj egy `Iteral` sablon eljárást, amely addig végez el egy elemen egy átalakítást újra meg újra, amíg valamilyen feltétel igazsá nem válik az elemre. Az elem típusa (korlátozott átlátszatlan), az átalakítás művelete és a megállás feltétele legyenek a sablon paraméterei!

2004. október 14.

Programozási nyelvek I. - 5. gyakorlat

15

Házi feladatok (2)

- Írj egy `BalInverz` sablon eljárást, amely egy félcsoporth tetszőleges eleméről megmondja, hogy van-e baloldali inverze, és ha van, akkor azt is visszaadja. A félcsoporth elemeinek típusa (diszkrét), a félcsoporth egységeleme és a szorzás művelete legyenek a sablon paraméterei. A szorzás műveletének legyen alapértelmezett értéke a szokásos műveleti jel. Példányosítsd egészekre összeadással!

2004. október 14.

Programozási nyelvek I. - 5. gyakorlat

16
