

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 diszkriminánsos rekord segítségével megvalósított verem csomagot úgy, hogy az elemek típusa, a verem megadható legnagyobb mérete és a diszkrimináns 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

1. feladat megoldása (1)

```
verem_dg.ads:
generic
  type Elem is private;
  Meret_Korlat : in Natural := 1000;
  Alap_Meret : in Natural := 100;
package Verem_DG is
  subtype Meret is Natural range
    1 .. Meret_Korlat;
  type Verem ( Max_Meret : Meret :=
    Alap_Meret ) is limited private;
  ...
end Verem_DG;
```

2004. október 14.

Programozási nyelvek I. - 5. gyakorlat

5

1. feladat megoldása (2)

```
verem_dg_proba.adb:
with Verem_DG, Text_IO;
use Text_IO;

procedure Verem_DG_Proba is
  package Int_Verem is
    new Verem ( Integer );
  use Int_Verem
  ...
end Verem_DG_Proba;
```

2004. október 14.

Programozási nyelvek I. - 5. gyakorlat

6

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. `Integer`)

2004. október 14.

Programozási nyelvek I. - 5. gyakorlat

7

2. feladat

Írj egy `Kiir` 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

2. feladat megoldása (1)

```
kiir.ads:
generic
  type T is limited private;
  with function To_String ( X : T )
    return String;

  procedure Kiir ( X : in T );
```

2004. október 14.

Programozási nyelvek I. - 5. gyakorlat

9

2. feladat megoldása (2)

```
kiir_proba.adb:
procedure Kiir_Proba is
  procedure Int_Kiir is
    new Kiir ( Integer, Integer'Image );
  begin
  end Kiir_Proba;
```

2004. október 14.

Programozási nyelvek I. - 5. gyakorlat

10

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 `Szuma` 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. Példányosítsd a sablont úgy, hogy produktumot számoljon!

2004. október 14.

Programozási nyelvek I. - 5. gyakorlat

12

3. feladat megoldása (1)

```
szumma.ads:
generic
  type Szam is private;
  type Index is (<>);
  type Vektor is array ( Index range <> )
                    of Szam;

  Zerus : in Szam;
  with function "+" ( A, B : Szam )
                return Szam;
function Szumma ( V : Vektor )
                return Szam;
```

2004. október 14.

Programozási nyelvek I. - 5. gyakorlat

13

3. feladat megoldása (2)

```
szumma_proba.ads:
...
procedure Szumma_Proba is
  type Int_Vektor is array
    ( Integer range <> ) of Integer;
  function Int_Prod is
    new Szumma ( Integer, Integer,
                Int_Vektor, 1, "*" );
...
end Szumma_Proba;
```

2004. október 14.

Programozási nyelvek I. - 5. gyakorlat

14

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