

Programozási nyelvek I. 3. gyakorlat

Balogh Ádám
bas@elte.hu

Eötvös Loránd Tudományegyetem
Informatikai Kar

Segítség az 1. feladathoz

- Altípus definiálása:
`subtype Név is Konstrukció;`
- Csak az eredeti típust módosító konstrukciók használhatók
- Egyirányú kompatibilitás az eredeti típussal
- Diszkrét típus megszorítása intervallumra
`Típus range Eleje .. Vége`

2004. szeptember 30.

Programozási nyelvek I. - 3. gyakorlat

4

1. feladat

Módosítsd az előző gyakorlaton írt `Naptar` programot úgy, hogy a `Positive` típus helyett annak egy megszorított altípusát használja, amely csak az 1 és 7 közötti számokat tartalmazza. Az új program neve legyen `Naptar2`!

2004. szeptember 30.

Programozási nyelvek I. - 3. gyakorlat

5

Segítség a 2. feladathoz

- Származtatott típus definiálása:
`type Név is new Konstrukció;`
- Csak az eredeti típust módosító konstrukciók használhatók, ugyanazok, mint altípus esetén
- Teljesen inkompatibilis az eredeti típussal

2004. szeptember 30.

Programozási nyelvek I. - 3. gyakorlat

7

2. feladat

Módosítsd az előbb megírt `Naptar2` programot úgy, hogy a `Positive` típus helyett egy belőle származtatott típust használjon, amely csak az 1 és 7 közötti számokat tartalmazza. Az új program neve legyen `Naptar3`!

2004. szeptember 30.

Programozási nyelvek I. - 3. gyakorlat

8

Segítség a 3. feladathoz

- Felsorolási típus:
`(Azonosító, Azonosító, ...)`
- A diszkrét típusok minden attribútuma alkalmazható rá

2004. szeptember 30.

Programozási nyelvek I. - 3. gyakorlat

10

3. feladat

Módosítsd az előbb megírt `Naptar3` programot úgy, `Positive`-ből származtatott típus helyett felsorolási használjon, amely a napok neveit tartalmazza. Az új program neve legyen `Naptar4`!

2004. szeptember 30.

Programozási nyelvek I. - 3. gyakorlat

11

Segítség a 4. feladathoz

- Vektor típus konstrukciója:

```
type Név is array ( Eleje .. Vége )
of Elemtípus;
```
- Teljes vektor megadása (pl. mint kezdőérték):

```
( Kifejezés, Kifejezés, ... )
```
- Vektor egy eleme: `Vektor (Index)`
- Egész számok típusa: `Integer`

2004. szeptember 30.

Programozási nyelvek I. - 3. gyakorlat

13

4. feladat

Írj programot, ami megkeresi egy 1-től 10-ig indexelt tömb elemeinek maximumát! A maximum helyét (indexét) a főprogramba ágyazott `Max` függvény adja vissza! A (konstans) tömb elemei legyenek 2, 6, 5, -9, 1, 56, 86, 23, -71 és 4. A program neve legyen `Max_Ker`!

2004. szeptember 30.

Programozási nyelvek I. - 3. gyakorlat

14

Segítség az 5. feladathoz

- Vektor típus konstrukciója egy típus teljes intervallumára:

```
type Név is array ( Indextípus )
of Elemtípus;
```
- Teljes vektor megadása indexszel jelölt formában (pl. mint kezdőérték):

```
( Indexhalmaz => Kifejezés,
  Indexhalmaz => Kifejezés, ... )
```
- Indexhalmaz: mint érték-halmaz a case-nél

2004. szeptember 30.

Programozási nyelvek I. - 3. gyakorlat

17

5. feladat

Alakítsd át az előző programot úgy, hogy a tömb indexeinek típusa eleve csak az 1 és 10 közötti számokat tartalmazza (altípus vagy származtatott típus)! A tömb elemei legyenek 2, 2, 0, 9, 9, 9, 4, 13, 4, 2, amit indexszel jelölt formában adj meg (még hozzá úgy, hogy egy érték csak egyszer szerepeljen)! Az új program neve legyen `Max_Ker2`!

2004. szeptember 30.

Programozási nyelvek I. - 3. gyakorlat

18

Segítség a 6. feladathoz (1)

- Tömb típus indexeinek intervallumát nyitva lehet hagyni, csak a típusát kell rögzíteni:

```
type Név is array ( Indextípus
  range <> ) of Elemtípus;
```
- Konkrét intervallum rögzítése változó deklarációjakkor:

```
Változó : Tömbtípus ( Intervallum );
```

2004. szeptember 30.

Programozási nyelvek I. - 3. gyakorlat

20

Segítség a 6. feladathoz (2)

- Tömb indextípusának `First` és `Last` attribútumai a tömbnek is attribútumai
- `X'First .. X'Last = X'Range`
- Teljes tömb megadásakor vegyes forma is használható (mint eljárás vagy függvényhívásnál paraméterátadásra)
- Itt is lehet `others`, mint a `case` ágainál

2004. szeptember 30.

Programozási nyelvek I. - 3. gyakorlat

21

6. feladat

Módosítsd az első `Max_Ker` programot úgy, hogy a tömb indexeinek intervallumát csak a konstans deklarációjakor rögzítéd! A tömb elemei most 2, 6, 4, 4, 4, 4, 4, 4, 4 és 4 legyenek, amit vegyes formában adj meg, és használd az `others`-et is! Az új program neve legyen `Max_Ker3`!

2004. szeptember 30.

Programozási nyelvek I. - 3. gyakorlat

22

Segítség a 7. feladathoz

- Nyitott index-intervallumú tömb típusú változók intervallumát a kezdőértékadás is rögzítheti (ha nincs benne `others`)
- Attribútumok segítségével lekérdezhető az ilyen módon automatikusan meghatározott intervallum

2004. szeptember 30.

Programozási nyelvek I. - 3. gyakorlat

25

7. feladat

Írd át a maximumkeresést (`Max_Ker`) úgy, hogy a tömb index-intervalluma nyitott legyen, mint az előző feladatnál, azonban most csak a változó kezdőértéke rögzítse! Az új program neve legyen `Max_Ker4`!

2004. szeptember 30.

Programozási nyelvek I. - 3. gyakorlat

26

Házi feladatok

- A legutolsó program `Max_Hely` függvényének segítségével írd egy rendező programot! A rendezés is a főprogramnak egy külön eljárása legyen! A program neve legyen `Rendezes`!
- Módosítsd az előző programot úgy, hogy ciklus helyett rekurziót használjon a rendezéshez! A program neve legyen `Rendezes2`!

2004. szeptember 30.

Programozási nyelvek I. - 3. gyakorlat

28
