Programozási nyelvek I.
4. gyakorlat

Balogh Ádám
bas@elte.hu
Eötvös Loránd Tudományegyetem
Informatikai Kar

Segítség az 1. feladathoz (1)

- Csomag specifikációja:
```
package Név is
   Deklarációk
end Név;
```
- Állomány neve: *név*.ads
- Csomag implementációja:
```
package body Név is
   Definíciók
end Név;
```
- Állomány neve: *név*.adb

Segítség az 1. feladathoz (2)

- Operátorok átlapolása: függvény neve "*operátor*" legyen, pl. "+"
- Lebegőpontos számok típusa:
```
Float digits Pontosság
```
- Többdimenziós tömb:
```
array ( Intervallum, Intervallum,
        ... ) of Típus
```
- Többdimenziós tömb attribútumai:
```
Attribútum ( Dimenzió )
```

1. feladat

Írj egy `Matrixok` nevű csomagot, amely a mátrixok típusát valósítja meg! A mátrixra legyen értelmezve az összeadás és a szorzás művelete, a szokásos műveleti jelekkel. A műveletek megvalósításánál feltételezhetjük, hogy az operandus mátrixok méretei megfelelőek a művelet elvégzéséhez. A főprogramot (`Matrixok_Proba`) töltsd le a gyakorlat honlapjáról!

Segítség a 2. feladathoz

- Rekord típus:

```
type Név is record
                Mezők
            end record;
```

- Mezők: mint változódeklarációk
- Átlátszatlan típus: private
- Átlátszatlan típus kifejtése: csomag specifikációjának privát részében, amit a private kulcsszó vezet be

2. feladat

Valósítsd meg a pozitív racionális számok típusát a `Poz_Rac` csomagban, átlátszatlan típusként! A típus reprezentációja egy számláló és nevező mezőket tartalmazó rekord legyen. Műveletek: két pozitív osztása, két pozitív racionális osztása, pozitív racionális osztása pozitívval, valamint számláló és nevező lekérdezése. A főprogramot (`Poz_Rac_Proba`) töltsd le a gyakorlat honlapjáról!

Segítség a 3. feladathoz

- Átlátszatlan típusokra a csomagban definiált műveleteken kívül értelmezve van az értékadás és az összehasonlítás is
- Korlátozott átlátszatlan típus (`limited private`): még ez a két művelet sincs értelmezve rá

3. feladat

Valósítsd meg a verem típust korlátozott átlátszatlan típusként a `Verem_V` csomagban. A műveletek legyenek: elem behelyezése, legfelső elem lekérdezése illetve törlése, valamint annak vizsgálata, hogy üres-e illetve tele van-e a verem. A főprogramot (`Verem_V_Proba`) töltsd le a gyakorlat honlapjáról!

Segítség a 4. feladathoz (1)

- Diszkriminánsos rekord típus:

```
type Név ( Diszkrimináns ) is record
                Mezők
            end record;
```

- A *`Diszkrimináns`* szintaxisa olyan, mint egy függvény paraméteréé
- Mezőkben paraméterként használhatjuk a diszkriminánsot
- Átlátszatlan típusnál is szerepelnie kell a diszkriminánsnak, nemcsak a kifejtésénél

Segítség a 4. feladathoz (2)

- Diszkrimináns a rekord egyik mezője, de értéke csak a teljes rekordnak való értékadáskor változtatható
- Diszkrimináns alapértelmezett értéke: ha a diszkriminánsot tömb méretére használjuk, és meghagyjuk az alapértelmezett értéket, akkor az indextípus által megengedett maximális méretű terület foglalódik le, de csak az alapértelmezett értéknyit lehet használni, amíg az egész új értéket nem kap

4. feladat

Módosítsd a vermet úgy, hogy méretét diszkrimináns adja meg! A diszkrimináns alapértelmezett értéke 100 legyen, és 1000-nél többet ne is lehessen megadni, ennyi foglalódjon le, ha az alapértelmezett értéket használjuk. A csomag neve legyen `Verem_D`!

Házi feladatok

- Egészítsd ki a pozitív racionális típust új műveletekkel: összeadás, kivonás, szorzás, valamint az összes művelet legyen végrehajtható pozitív racionális és pozitív között (pozitív racionális eredményt adva)!
- Valósítsd meg a sor típust a (továbbfejlesztett) verem típushoz hasonlóan!