

Programozási nyelvek I. 10. gyakorlat

Balogh Ádám
bas@elte.hu

Eötvös Loránd Tudományegyetem
Informatikai Kar

Segítség az 1. feladathoz (1)

- Szelektív várakoztatás:

```
select
    ág
or
    ág
or
    ...
end select;
```
- Egy ág lehet feltételes is:

```
when Feltétel => ág
```

2004. december 2.

Programozási nyelvek I. - 10. gyakorlat

2

Segítség az 1. feladathoz (2)

- Adott pillanatban egy ág nyitott, ha nincs feltétele, vagy a feltétele igaz
- Csak nyitott ágba tud belépni
- Ha minden ág zárt, kivétel keletkezik
- Ágak többsége egy `accept` utasítással (randevú) kezdődik
- Lezáró ág: `terminate` utasítással kezdődik, akkor hívódik meg, ha az `accept` ágakat már semelyik taszk nem tudja hívni

2004. december 2.

Programozási nyelvek I. - 10. gyakorlat

3

Segítség az 1. feladathoz (3)

- Lehet `else` ág is, amit `or` helyett `else` kulcsszó vezet be, amit közönséges utasítássorozat követ: akkor választódik ki, ha minden ág zárt
- Feltételes belépési pont hívás:

```
select
    Hívás
else
    Utasítások
end select;
```

2004. december 2.

Programozási nyelvek I. - 10. gyakorlat

4

1. feladat

Írj kocsmá szimulátort! A részek (taszkek) a kocsmába egy ajtón keresztül tudnak bemenni. A kocsmában csak öt ember fér el, ennél több nem mehet be az ajtón, azonban hét részeg próbál bejutni. Aki bejutott, az megiszik egy sört (10 mp), és hazamegy. Aki nem jut be, az előbb alszik egyet a parkban (5 mp), majd megpróbál újra bejutni. A részek jelenítsék meg, hogy épp isznak, vagy elmentek aludni. A program neve legyen `Kocsmá`!

2004. december 2.

Programozási nyelvek I. - 10. gyakorlat

5

1. feladat megoldása (1)

```
kocsmá.adb:
...
task body Ajtó is
    Benn : Natural := 0;
begin
    loop
        select
            accept Kilep;
            Benn := Benn - 1;
        or when Benn < 5 =>
            accept Belep;
        (folyt. köv.)
```

2004. december 2.

Programozási nyelvek I. - 10. gyakorlat

6

1. feladat megoldása (2)

```
kocsmas.adb:
(folyt.)
    Benn := Benn + 1;
    or
    terminate;
  end select;
end loop;
end Ajto;
task body Reszeg is
begin
  loop
(folyt. köv.)
```

2004. december 2.

Programozási nyelvek I. - 10. gyakorlat

7

1. feladat megoldása (3)

```
kocsmas.adb:
(folyt.)
    select
      Ajto.Belep;
      Put_Line ( "Reszeg iszik." );
      delay 10.0;
      Ajto.Kilep;
      exit;
    else
      Put_Line ( "Reszeg alszik." );
      delay 5.0;
    ...
```

2004. december 2.

Programozási nyelvek I. - 10. gyakorlat

8

Segítség a 2. feladathoz

- Várakozó ág: `delay` utasítással kezdődik, és akkor hajtódik végre, ha egyik nyitott `accept` ágat sem hívják
- Ha várakozás közben meghívják az egyik `accept` ágat, akkor a várakozás megszakad, és a meghívott ág hajtódik végre
- Időhöz kötött hívás: hasonló a feltételes híváshoz, de `else` ág helyett `or` ág van, ami `delay` utasítással kezdődik

2004. december 2.

Programozási nyelvek I. - 10. gyakorlat

9

2. feladat

Egészítsd ki a programot egy kocsmáros taszkkal! A kocsmáros háromféle italt tölt: sört, bort és pálinkát. A töltés italfajtánként rendre 1, 0,1 és 0,2 mp-ig tart. Ha 5 mp-ig senki nem kér semmit, akkor a kocsmáros hazamegy. A részek belépés után felhajtanak egy pálinkát (0,2 mp), megisznak egy pohár bort (1 mp), majd végtelen sörözésbe kezdenek (első 2 mp, többi mindig 2 mp-cel hosszabb, mint az előző). Az új program neve legyen `Kocsmas2`!

2004. december 2.

Programozási nyelvek I. - 10. gyakorlat

10

2. feladat megoldása (1)

```
kocsmas2.adb:
...
  select
    accept Tolt ( Mit : in Ital ) do
      Put_Line ( "K: " & Ital'Image (
        Mit ) & " toltok" );
      delay ( Toltido ( Mit );
    end Tolt;
  or
    delay ( Zaras_Varas );
    exit;
(folyt. köv.)
```

2004. december 2.

Programozási nyelvek I. - 10. gyakorlat

11

2. feladat megoldása (2)

```
kocsmas2.adb:
...
  loop
    ...
  end loop;
exception
  when Tasking_Error =>
    Put_Line ( "Reszeg: Bezart?" );
    Ajto.Kilep;
  ...
```

2004. december 2.

Programozási nyelvek I. - 10. gyakorlat

12

Segítség a 3. feladathoz

- Taszkok diszkriminánása csak diszkrét típus vagy mutató lehet
- Taszknak is lehet mutatója
- Taszk létrehozása dinamikusan: new utasítással, az utasítás végrehajtásakor jön létre
- Dinamikusan létrehozott taszkat akkor szabadítja fel a szeméthyűjtő, ha sehonnan nem mutatnak rá, és véget ért

2004. december 2.

Programozási nyelvek I. - 10. gyakorlat

13

3. feladat

Egészítsd ki a kocsmát egyetemistákkal! Ők csak egy pohár bort isznak meg (3 mp). Ha nem jutnak be a kocsmába 1 mp alatt, inkább elmennek analízis előadásra. Három ismeretlen egyetemista a részekkel együtt próbál bemenni, Janics 2 mp-cel később, Juliska pedig újabb 1 mp múlva. A kiegészített program neve legyen `Kocsm3`!

2004. december 2.

Programozási nyelvek I. - 10. gyakorlat

14

3. feladat megoldása

```
kocsm3.adb:
...
task type Egyetemista ( Nev : String_Mut :=
                        null );
type Egyetemista_Mut is access Egyetemista;
...
Egy_Egyetemista := new Egyetemista (
                        new String' ( "Janics" );
...

```

2004. december 2.

Programozási nyelvek I. - 10. gyakorlat

15

Segítség a 4. feladathoz (1)

- Egy védett egységeknek is lehet privát része és adattagjai is (mint egy csomagnak)
- Védett egységekben lehetnek függvények is, de két függvény között nincs kölcsönös kizárás
- Belépési pontok védet egységekben: mint eljárások, de lehet őrfeltételük, és várakozási sor tartozik hozzájuk
- Belépési pontok specifikációja: mint taszkok belépési pontjaié

2004. december 2.

Programozási nyelvek I. - 10. gyakorlat

16

Segítség a 4. feladathoz (2)

- Belépési pont implementációja:


```
entry Név ( Paraméterek )
when Feltétel is
    ...
```
- Figyelem! Eljárások és belépési pontok nem hívhatják sem egymást, sem függvényt ugyanazon a védett egységen belül
- Vezérlésátadás belépési pontok között:


```
requeue Pontnév ( Paraméterek );
```

2004. december 2.

Programozási nyelvek I. - 10. gyakorlat

17

4. feladat

Írd át az ajtó taszkat védett egységgé. A belépés egy belépési ponton, a kilépés pedig egy eljáráson keresztül történjen, valamint egy függvénnyel le lehessen kérdezni a kocsmá telítettségét is (lebegőpontos szám)! A módosított program neve legyen `Kocsm4`!

2004. december 2.

Programozási nyelvek I. - 10. gyakorlat

18

4. feladat megoldása (1)

```
kocsmas4.adb:
...
protected Ajto is
  entry Belep;
  procedure Kilep;
  function Telitettseg return Float;
private
  Max : Natrual := 5;
  Benn : Natural := 0;
end Ajto;
(folyt. köv.)
```

2004. december 2.

Programozási nyelvek I. - 10. gyakorlat

19

4. feladat megoldása (2)

```
kocsmas4.adb:
(folyt.)
protected body Ajto is
  entry Belep when Benn < Max is
  begin
    Benn := Benn + 1;
  end Belep;
  ...
end Ajto;
...
```

2004. december 2.

Programozási nyelvek I. - 10. gyakorlat

20

Házi feladat

Írj tanár taszkt a kocsmába! Ha a kocsmában a telítettség 0,8 alatt van, akkor bemegy, és megiszik egy pohár bort, és utána (kis késéssel) elmegy analízis előadást tartani, különben pedig nem késik. A végső program neve legyen Kocsmas5!

2004. december 2.

Programozási nyelvek I. - 10. gyakorlat

21