Programozási nyelvek I.
9. gyakorlat

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

Segítség az 1. feladathoz

- Versenyhelyzet: két taszk ugyanahhoz a kizárólagos erőforráshoz próbál meg hozzáférni úgy, hogy legalább az egyik változtatni is próbál rajta
- Kizárólagos erőforrások: változók, állományok (beleértve a szabványos kimenetet és bemenetet is)

1. feladat

Írj egy saját karakterlánc kiíró eljárást amely karakterenként ír ki egy karakterláncot, és ezt használd a legutóbbi `Jancsi_Julcsa` programban a `Put_Line` helyett! Az új program neve legyen `Jancsi_Julcsa4`! Figyeld meg a futás eredményét! (Ha túl gyors, irányítsd át adatállományba!)

Segítség a 2. feladathoz

- Kritikus szakasz: az a kódrészlet, amelyben a taszk a közös, kizárólagos erőforráshoz nyúl
- Egyszerre egy taszk lehet a kritikus szakaszban
- *Dijkstra*-féle szemaforok bináris változata: `P` művelet (*proberen*) jelzi a belépési kísérletet a kritikus szakasz, a `V` művelet (*verhogen*) pedig a kilépést onnan

2. feladat

Javítsd ki az előző programot egy `Szemafor` nevű taszk segítségével, ami egy Dijkstra-féle szemafor bináris változatát valósítja meg! A javított program neve legyen `Jancsi_Julcsa5`! Figyeld meg ismét a futás eredményét!

Segítség a 3. feladathoz (1)

- Védett egység: egyszerre csak egy eljárás lehet benne aktív
- Taszkhoz hasonlóan nem önálló fordítási egység
- Specifikáció:

```
protected vagy protected type Név is
   Deklarációk
end Név;
```

Segítség a 3. feladathoz (2)

- Törzs:

```
protected body Név is
   Definíciók
end Név;
```

- Taszkokhoz hasonlóan itt is lehet a törzs `separate`

3. feladat

Javítsd ki az `Jancsi_Julcsa4` programot egy `Vedett` nevű védett objektum segítségével, ami a `Kiir` eljárást tartalmazza! A javított program neve legyen `Jancsi_Julcsa6`! Figyeld meg most is a futás eredményét!