

## Programozási nyelvek I. 1. gyakorlat

Balogh Ádám  
bas@elte.hu

Eötvös Loránd Tudományegyetem  
Informatikai Kar

## Információk a tárgyról

- Gyakorlat: Csütörtök 12:00-13:30  
Déli tömb 00-803  
Balogh Ádám
- Web: <http://bas.web.elte.hu/adagyak1>
- Előadás: Szerda 8:00-10:00  
Északi tömb Konferenciaterem  
Kozsik Tamás

2004. szeptember 16.

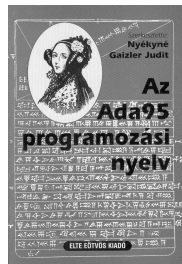
Programozási nyelvek I. - 1. gyakorlat

2

## Ajánlott irodalom

Nyékyné Gaizler Judit

**Az  
Ada95  
programozási nyelv**  
ELTE Eötvös Kiadó  
Budapest  
1998



2004. szeptember 16.

Programozási nyelvek I. - 1. gyakorlat

3

## Követelmények

- Legalább 2 db legalább elégséges zárthelyi
- Minden zárthelyi előtt 2-2 db beadandó feladat
- Minden gyakorlaton +/- a legutóbbi előadás anyagából

2004. szeptember 16.

Programozási nyelvek I. - 1. gyakorlat

4

## Javítási lehetőségek

- Pótzárthelyi a félév végén
- Pótteszt a félév végén
- Beadandók pótolhatók a következő zárthelyiig (de ekkor az aktuális zárthelyi nem írható meg)

2004. szeptember 16.

Programozási nyelvek I. - 1. gyakorlat

5

## Értékelés

- Ha a +/-ok összege a félév végére legalább 0, vagy a pótteszt sikeres, és van két sikeres zárthelyi: gyakorlati jegy a két legjobb eredmény átlaga
- Ha sem a +/-ok összege nem éri el a 0-t, sem a pótteszt nem sikeres, de két zárthelyi igen, és átlaguk legalább közepes, akkor ez a következő félévre átvihető gyakorlati jegynek

2004. szeptember 16.

Programozási nyelvek I. - 1. gyakorlat

6

## A gyakorlatok menete

- A gyakorlatok elején +/-!
- Feladatok megjelennek a weboldalon
- Később a megoldásuk is megjelenik
- Házi feladatok: nem kötelező, de ajánlott

2004. szeptember 16.

Programozási nyelvek I. - 1. gyakorlat

7

## Beadandók beadása

- Beadandókat a `caesar.elte.hu` gépen kell bemásolni a következő könyvtárba:  
`/afs/elte.hu/user/b/bas/public/adagyak/beadN/ETR-azonosító`
- *N* értelemszerűen 1, 2, 3 vagy 4, az *ETR-azonosító* pedig mindenki saját ETR azonosítójának első 7 betűje
- A bármi más módon beadni próbált beadandókat figyelmen kívül hagyom!

2004. szeptember 16.

Programozási nyelvek I. - 1. gyakorlat

8

## Bevitel, fordítás, szerkesztés, futtatás

- **Ada fordító:** GNAT (GNU Ada Translator)
- **Bevitel:** tetszőleges editor (pl. (x)emacs, joe, Notepad stb.) vagy (csak Windows-on): AdaGIDE
- **Forrásállomány neve:** `programnév.adb`
- **Fordítás:** `gnatmake programnév`
- **Futtatás:** mint minden programot: Unix-on: `./programnév`, Windows-on: `programnév`

2004. szeptember 16.

Programozási nyelvek I. - 1. gyakorlat

9

## Segítség az 1. feladathoz

- Főprogram:  

```
procedure Programnév is
begin
    Legalább 1 utasítás
end Programnév;
```
- Üres utasítás: `null`
- Minden utasítást pontosvessző zár!
- Programnév ugyanaz, mint az állomány neve, kis és nagybetű ugyanaz

2004. szeptember 16.

Programozási nyelvek I. - 1. gyakorlat

10

## 1. feladat

Írj egy működő Ada programot, ami semmit sem csinál! A program neve legyen: `Semmi`!

2004. szeptember 16.

Programozási nyelvek I. - 1. gyakorlat

11

## 1. feladat megoldása

```
semmi.adb:
procedure Semmi is
begin
    null;
end Semmi;
```

2004. szeptember 16.

Programozási nyelvek I. - 1. gyakorlat

12

## Segítség a 2. feladathoz

- Csomag használata: program elején (még a `procedure` előtt): `with Csomagnév;`
- Csomagbeli eljárások használata: `Csomagnév.Eljárásnév` vagy ha a `procedure` előtt `use Csomagnév` is van, akkor csak `Eljárásnév`
- Szövegek kiírása: `Text_IO` csomag `Put_Line` eljárása
- Használata: `Put_Line ( "Szöveg" );`

2004. szeptember 16.

Programozási nyelvek I. - 1. gyakorlat

13

## 2. feladat

Írj egy működő Ada programot, ami a „Hello Világ!” szöveget írja a képernyőre! A program neve legyen: `Hello!`

2004. szeptember 16.

Programozási nyelvek I. - 1. gyakorlat

14

## 2. feladat megoldása

```
hello.adb:
with Text_IO;
use Text_IO;

procedure Hello is
begin
  Put_Line ( "Hello Világ!" );
end Hello;
```

2004. szeptember 16.

Programozási nyelvek I. - 1. gyakorlat

15

## Segítség a 3. feladathoz (1)

- Változók deklarálása: `begin` elé:  
`Név : Típus;`
- Kezdőérték adással együtt:  
`Név : Típus := Érték;`
- Több változó együtt:  
`Egyik, Másik : Típus [:= Érték]`
- Természetes számok típusa: `Natural`
- Értékadás művelete: `Változó := Érték`

2004. szeptember 16.

Programozási nyelvek I. - 1. gyakorlat

16

## Segítség a 3. feladathoz (2)

- Alapműveletek: `+`, `-`, `*`, `/`
- Számlálós ciklus:  
`for Változó in Eleje .. Vége loop`  
`Utasítások`  
`end loop`
- Karakterláncok összekapcsolása: `"Egyik"` & `"Másik"`
- Természetes szám konvertálása karakterláncná: `Natural'Image ( Érték )`

2004. szeptember 16.

Programozási nyelvek I. - 1. gyakorlat

17

## 3. feladat

Írj Ada programot, amely kiszámítja 10 faktoriálisát, és kiírja a képernyőre! Az eredmény elé a program írja ki, hogy „10! =” A program neve legyen: `Faktor!`

2004. szeptember 16.

Programozási nyelvek I. - 1. gyakorlat

18

### 3. feladat megoldása

```
faktor.adb:
procedure Faktor is
  F : Natural := 1;
begin
  for I in 1 .. 10 loop
    F := F * I;
  end loop;
  Put_Line ( "10!=" &
             Natural'Image ( F ));
end Faktor;
```

2004. szeptember 16.

Programozási nyelvek I. - 1. gyakorlat

19

### Segítség a 4. feladathoz (1)

- További típusok: Character (karakter), String (karakterlánc) és Boolean (logikai)
- Konstansok definiálása:  
Név : constant Típus := Érték
- Összehasonlítás: =, /=, <, <=, >, >=
- Logikai operátorok: not, and és or
- Rövidzáras logikai operátorok: and then és or else

2004. szeptember 16.

Programozási nyelvek I. - 1. gyakorlat

20

### Segítség a 4. feladathoz (2)

- Logikai konstansok: True és False
- Karakterlánc elemei:  
Karakterlánc ( Index )
- Első és utolsó elem indexe:  
Karakterlánc'First és  
Karakterlánc'Last
- Elöltesztelt, feltételvezérelt ciklus:  
while Feltétel loop  
    Utasítások  
end loop

2004. szeptember 16.

Programozási nyelvek I. - 1. gyakorlat

21

### Segítség a 4. feladathoz (3)

- Kétágú elágazás:  
if Feltétel then  
    Utasítások  
else  
    Utasítások  
end if

2004. szeptember 16.

Programozási nyelvek I. - 1. gyakorlat

22

### 4. feladat

Írj Ada programot, amely megkeresi a „Hello Világ!” szövegben az „o” betűt, és a keresés eredményétől függően kiírja a képernyőre, hogy „Megvan.” vagy hogy „Nincs meg.”! A program neve legyen: Lin\_Ker!

2004. szeptember 16.

Programozási nyelvek I. - 1. gyakorlat

23

### 4. feladat megoldása (1)

```
lin_ker.adb:
...
procedure Lin_Ker is
  S : constant String := "Hello Világ";
  C : constant Character := 'o';
  P : Natural := S'First - 1;
  V, L : Boolean := False;
begin
  while ( not V ) and then ( not L ) loop
    V := ( P + 1 >= S'Last );
    (folyt. köv.)
```

2004. szeptember 16.

Programozási nyelvek I. - 1. gyakorlat

24

#### 4. feladat megoldása (2)

```
lin_ker.adb:
(folyt.)
  L := ( S ( P + 1 ) = C );
  P := P + 1;
end loop;
if L then
  Put_Line ( "Megvan." );
else
  Put_Line ( "Nincs meg." );
end if;
end Lin_Ker;
```

2004. szeptember 16.

Programozási nyelvek I. - 1. gyakorlat

25

#### Házi feladatok

- Írj programot, amely egy rendezett karakterláncban megkeres egy adott karaktert logaritmikus keresés segítségével. A program neve legyen `Log_Ker`!
- Javítsd ki a gyakorlaton megírt lineáris keresést úgy, hogy üres karakterláncokra is működjön (ha eddig nem működött). Az új program neve legyen `Lin_Ker2`!

2004. szeptember 16.

Programozási nyelvek I. - 1. gyakorlat

26