

An aerial photograph of Budapest, Hungary, showing the city's layout along the Danube River. The image is in black and white, with a semi-transparent white rectangular box in the center containing the title text. The city's architecture, including domes and spires, is visible, as is the river and surrounding landscape.

Programozási alapismeretek 1. előadás

Tartalom

- A problémamegoldás lépései –
programkészítés folyamata
- A specifikáció
- Az algoritmus
- Algoritmikus nyelvek –
struktogram
- A kódolás –
a fejlesztői környezet



A programkészítés folyamata



1. **Specifikálás** (miből?, mit?) → *specifikáció*
2. **Tervezés** (mivel?, hogyan?) → *adat- + algoritmus-leírás*
3. **Kódolás** (a gép hogyan?) → *kód* (reprezentáció + implementáció)
4. **Tesztelés** (hibás-e?) → *hibalista* (diagnózis)
5. **Hibakeresés** (hol a hiba?) → *hibahely, -ok*
6. **Hibajavítás** (hogyan jó?) → *helyes program*
7. **Minőségvizsgálat, hatékonyság** (jobbítható-e?, hogyan?) → *jó program*
8. **Dokumentálás** (hogyan működik, használható?) → *használható program*
9. **Használat, karbantartás** (még mindig jó?) → *évelő (időtálló) program*



„Interface” a megbízó
és a fejlesztő között

A specifikáció fogalma

Célja:

a feladat formális megragadása.

Összetevői:

1. Bemenő adatok (azonosító, értékhalmaz [mértékegység])
2. Ismeretek a bemenetről (előfeltétel)
3. Eredmények (azonosító, értékhalmaz)
4. Az eredményt meghatározó állítás (utófeltétel)
5. A használt fogalmak definíciói
6. A megoldással szembeni követelmények
7. Korlátozó tényezők



A specifikáció fogalma

Tulajdonságai:

1. „Egyértelmű”, pontos, teljes
2. Rövid, tömör; formalizált
3. Szemléletes, érthető (fogalmak)

Specifikációs eszközök:

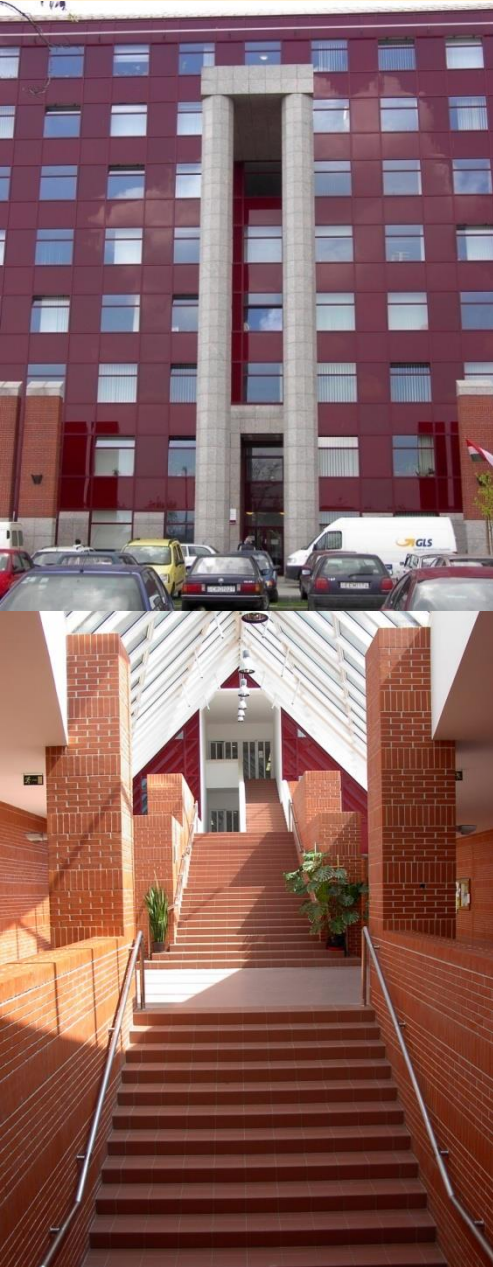
1. Szöveges leírás
2. Matematikai megadás



Az algoritmus fogalma

Az algoritmusok összeállítási módjai:

- Szekvencia (egymás utáni végrehajtás)
- Elágazás (választás 2 vagy több tevékenységből)
- Ciklus (ismétlés adott darabszámszor vagy adott feltételtől függően)
- Alprogram (egy összetett tevékenység, egyedi néven – absztrakció)



Példa: háromszög (specifikáció)

Feladat:

3 szám lehet-e egy derékszögű háromszög 3 oldala?

Specifikáció:

➤ **Bemenet:** $x, y, z \in \mathbb{R}$

➤ **Kimenet:** $\text{lehet} \in \mathbb{L}$

➤ **Előfeltétel:** $x > 0$ és $y > 0$ és $z > 0$

➤ **Utófeltétel:** $\text{lehet} = (x^2 + y^2 = z^2)$

Megjegyzés: a 3 szám sorrendjét ezek szerint rögzítettük – z az átfogó hossza!

\mathbb{R} = Valós számok halmaza

\mathbb{L} = Logikai értékek halmaza



Példa: háromszög (specifikáció)

Specifikáció = függvény:

Független változók

➤ **Bemenet:** $x, y, z \in \mathbb{R}$

a függvény értelmezési tartománya: $\mathbb{R} \times \mathbb{R} \times \mathbb{R} = \mathbb{R}^3$
(amelynek egyes komponenseire lehet hivatkozni a specifikációban x -szel, y -nal, z -vel)

Függő változó

➤ **Kimenet:** lehet $\in \mathbb{L}$

a függvény értékkészlete: \mathbb{L} (amelyre hivatkozhatunk a specifikációban lehet-tel)

➤ **Előfeltétel:** $x > 0$ és $y > 0$ és $z > 0$

a függvény értelmezési tartományának (\mathbb{R}^3) szűkítése (\mathbb{R}_+^3)

➤ **Utófeltétel:** lehet $= (x^2 + y^2 = z^2)$

mi igaz a végeredményre



Példa: háromszög (algorithmus)

Algorithmus:

A programunk 4 fő részből áll:
az adatok **deklarálása**, **beolvasása**, az
eredmény **kiszámítása**, az eredmény
kiírása:

Specifikáció:

- > **Bemenet:** $x, y, z \in \mathbb{R}$
- > **Kimenet:** lehet $\in L$
- > **Előfeltétel:** $x > 0$ és $y > 0$ és $z > 0$
- > **Utófeltétel:** lehet $= (x^2 + y^2 = z^2)$

Be: x, y, z [$x > 0$ és $y > 0$ és $z > 0$]

lehet $:= (x^2 + y^2 = z^2)$

Ki: lehet

Változó

x, y, z : **Valós**
lehet: **Logikai**

A **deklarációt**, az „elemi” utasításokat egy-egy
„dobozba” írjuk.

Később a be- és kimenetet nem algoritmizáljuk!

Példa: háromszög (algorithmus)

Algorithmus:

A programunk 4 fő részből áll:
az adatok **deklarálása**, **beolvasása**, az
eredmény **kiszámítása**, az eredmény
kiírása:

Specifikáció:

- > **Bemenet:** $x, y, z \in \mathbb{R}$
- > **Kimenet:** lehet $\in L$
- > **Előfeltétel:** $x > 0$ és $y > 0$ és $z > 0$
- > **Utófeltétel:** lehet $= (x^2 + y^2 = z^2)$

Valós: Valós számok **típusa**
Logikai: Logikai értékek **típusa**

Be: x, y, z $[x > 0 \text{ és } y > 0 \text{ és } z > 0]$

lehet $:= (x^2 + y^2 = z^2)$

Ki: lehet

Változó

x, y, z : **Valós**
lehet: **Logikai**

A **deklarációt**, az „elemi” utasításokat egy-egy „dobozba” írjuk.

Később a be- és kimenetet nem algoritmizáljuk!

Példa: háromszög (algorithmus)

Egy másik **algorithmus** a lényegi részre:

Segéd változók deklarálása
„széljegyzetként”

Változó
xx,yy,zz:Valós

xx:=x²

yy:=y²

zz:=z²

lehet:=(xx+yy=zz)

Bevezethető/-endők segéd (belső, saját)
változók.



Példa: másodfokú egyenlet (specifikáció)

Feladat:

Adjuk meg a másodfokú egyenlet egy megoldását! Az egyenlet: $ax^2+bx+c=0$

Kérdések:

- Mitől függ a megoldás? – *bemenet*
- Mi a megoldás? – *kimenet*
- Mit jelent: „megoldásnak lenni”? – *utófeltétel*
- Mindig/Mikor *van* megoldás? – *előfeltétel*
- Biztos *egy* megoldás van? – *kimenet/utófeltétel*



Példa: másodfokú egyenlet (specifikáció)

Specifikáció₁:

- Bemenet: $a, b, c \in \mathbb{R}$
- Kimenet: $x \in \mathbb{R}$
- Előfeltétel: –
- Utófeltétel₁: $ax^2 + bx + c = 0$

Megjegyzés: az uf. nem ad algoritmizálható információt. Nem baj, sőt tipikus, de ... próbálkozzunk még!



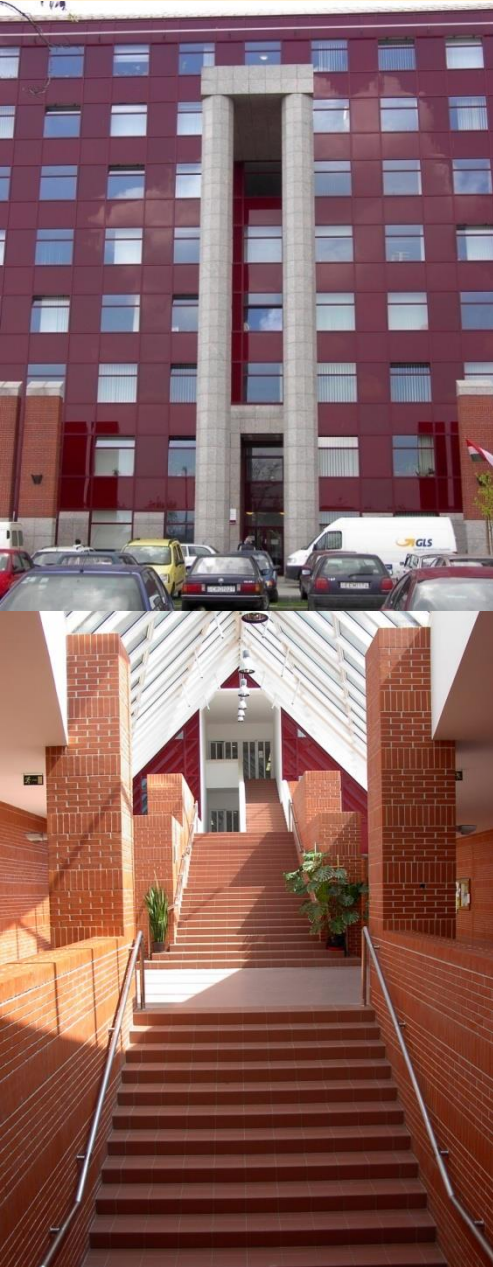
Példa: másodfokú egyenlet (specifikáció)

Specifikáció₁:

- Bemenet: $a, b, c \in \mathbb{R}$
- Kimenet: $x \in \mathbb{R}$
- Előfeltétel: –
- Utófeltétel₁: $ax^2 + bx + c = 0$

Megjegyzés: az uf. nem ad algoritmizálható információt. Nem baj, sőt tipikus, de ... próbálkozzunk még!

Megoldóképlet:
$$x_{1,2} = \frac{-b \pm \sqrt{b^2 - 4 * a * c}}{2 * a}$$



Példa: másodfokú egyenlet (specifikáció)

Specifikáció₂:

➤ Bemenet: $a, b, c \in \mathbb{R}$

➤ Kimenet: $x \in \mathbb{R}$

➤ Előfeltétel: $a \neq 0$

➤ Utófeltétel₂: $x = \frac{-b + \sqrt{b^2 - 4 * a * c}}{2 * a}$

Nyitott kérdések:

➤ *Mindig/ Mikor van megoldás?*

➤ *Egy megoldás van?*



Példa: másodfokú egyenlet (specifikáció)

Specifikáció₂:

➤ Bemenet: $a, b, c \in \mathbb{R}$

➤ Kimenet: $x \in \mathbb{R}$

➤ Előfeltétel: $a \neq 0$

➤ Utófeltétel₂: $x = \frac{-b + \sqrt{b^2 - 4 * a * c}}{2 * a}$

Nyitott kérdések:

➤ *Mindig/ Mikor van megoldás?*

➤ *Egy megoldás van?*



Példa: másodfokú egyenlet (specifikáció)

A „feladat-függvény”
értékkészlete: $\mathbb{R} \times \mathbb{L}$

Kimenet bővítés:

- Kimenet: $x \in \mathbb{R}$, $van \in \mathbb{L}$
- Utófeltétel: $van = (b^2 - 4 * a * c \geq 0)$ és

$$van \rightarrow x = \frac{-b + \sqrt{b^2 - 4 * a * c}}{2 * a}$$

Nyitott kérdés:

- *Egy megoldás van? – hf.*



Példa: másodfokú egyenlet (algoritmus)

Algoritmus:

Specifikáció₂:

- Bemenet: $a, b, c \in \mathbb{R}$
- Előfeltétel: $a \neq 0$
- Kimenet: $x \in \mathbb{R}$, $\text{van} \in \mathbb{L}$
- Utófeltétel: $\text{van} = (b^2 - 4ac \geq 0)$ és

$$\text{van} \rightarrow x = \frac{-b + \sqrt{b^2 - 4ac}}{2a}$$



Be: a, b, c [$a \neq 0$]

$d := b^2 - 4ac$

$\text{van} := d \geq 0$

van?

...

Változó

a, b, c, x : **Valós**

van : **Logikai**

d : **Valós**

Példa: másodfokú egyenlet (algorithmus)

Algorithmus:

Specifikáció₂:

- Bemenet: $a, b, c \in \mathbb{R}$
- Előfeltétel: $a \neq 0$
- Kimenet: $x \in \mathbb{R}$, $\text{van} \in \mathbb{L}$
- Utófeltétel: $\text{van} = (b^2 - 4 * a * c \geq 0)$ és

$$\text{van} \rightarrow x = \frac{-b + \sqrt{b^2 - 4 * a * c}}{2 * a}$$

Változó

a, b, c, x : Valós

van : Logikai

d : Valós

Be: a, b, c [$a \neq 0$]

$d := b^2 - 4 * a * c$

$\text{van} := d \geq 0$

I / van? / N

Igaz-ág
 $x := \frac{-b + \sqrt{d}}{2 * a}$

Hamis-ág

...



Példa: másodfokú egyenlet (algorithmus)

Algorithmus:

Specifikáció₂:

- Bemenet: $a, b, c \in \mathbb{R}$
- Előfeltétel: $a \neq 0$
- Kimenet: $x \in \mathbb{R}$, $\text{van} \in \mathbb{L}$
- Utófeltétel: $\text{van} = (b^2 - 4 * a * c \geq 0)$ és

$$\text{van} \rightarrow x = \frac{-b + \sqrt{b^2 - 4 * a * c}}{2 * a}$$

Változó

a, b, c, x : Valós

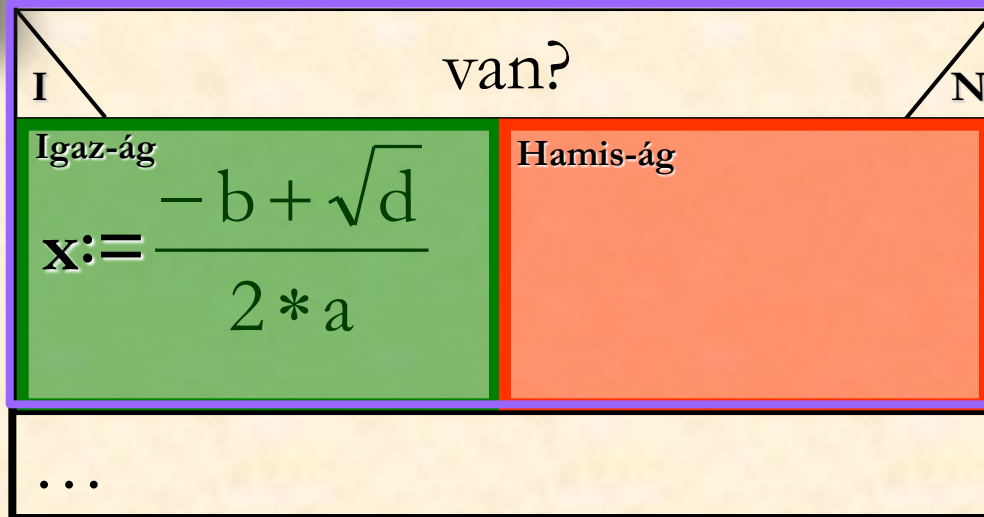
van : Logikai

d : Valós

Be: a, b, c [$a \neq 0$]

$d := b^2 - 4 * a * c$

$\text{van} := d \geq 0$



A **feltételes** utasítás „3-dobozos” struktúra.



Példa: másodfokú egyenlet (algoritmus)

Algoritmus másképpen:

Program MásodfokúEgyenlet:

Változó

a, b, c, x : Valós

van: Logikai

d : Valós

Be: a, b, c [$a \neq 0$]

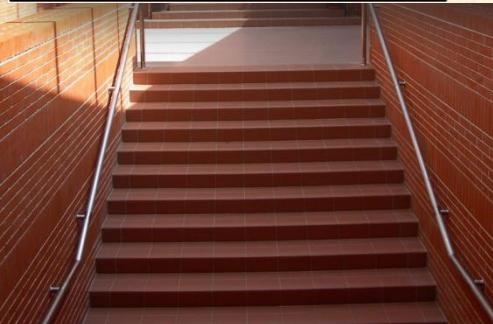
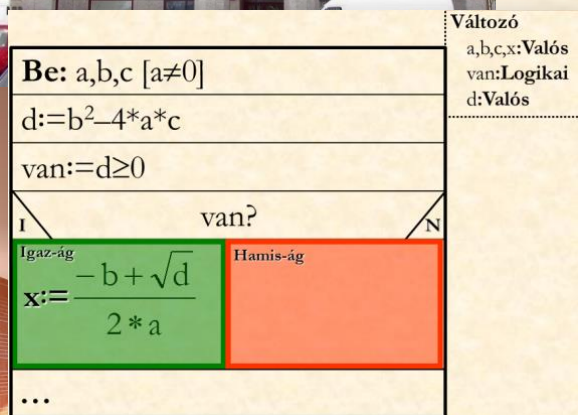
$d := b^2 - 4 * a * c$

van := $d \geq 0$

Ha van **akkor** $x := \frac{-b + \sqrt{d}}{2 * a}$

...

Program vége.



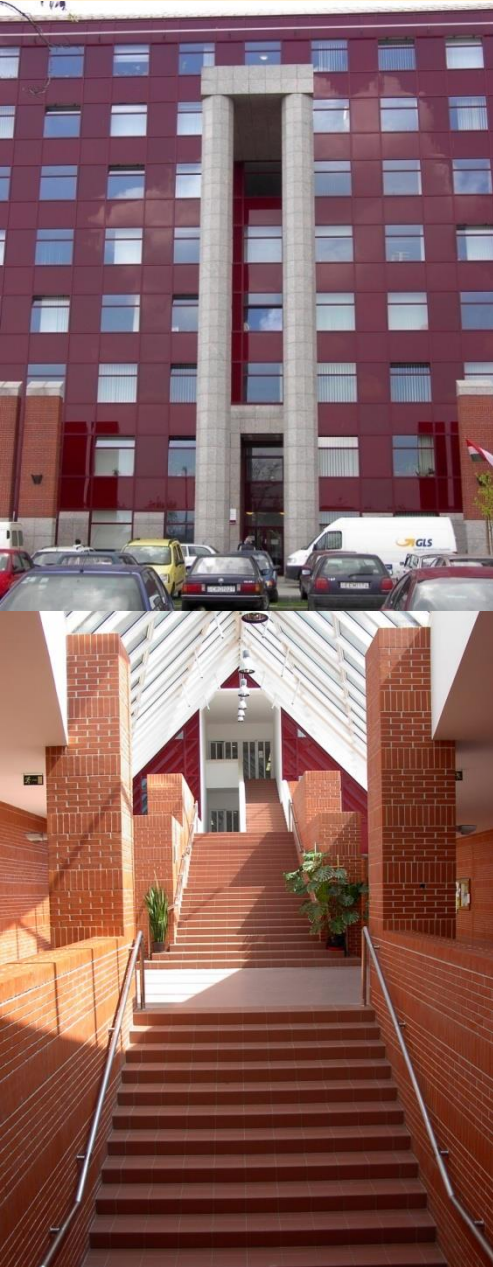
Algoritmusleíró nyelvek

➤ Szöveges leírás

- Mondatokkal leírás
- Mondatszerű elemekkel – **pseudokód**

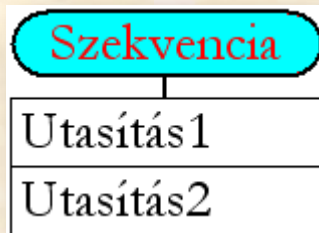
➤ Rajzos leírás

- Folyamatábra
- **Struktogram**

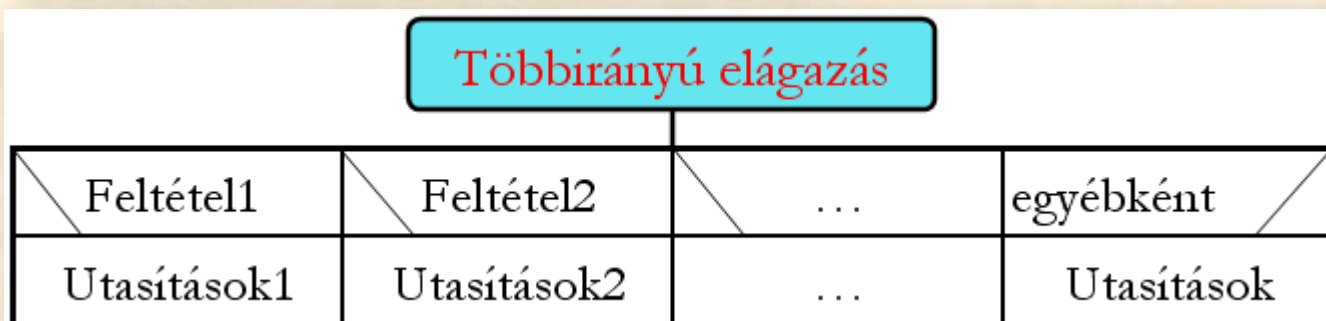


Struktogram (és pseudokód)

➤ Szekvencia:



➤ Elágazások:



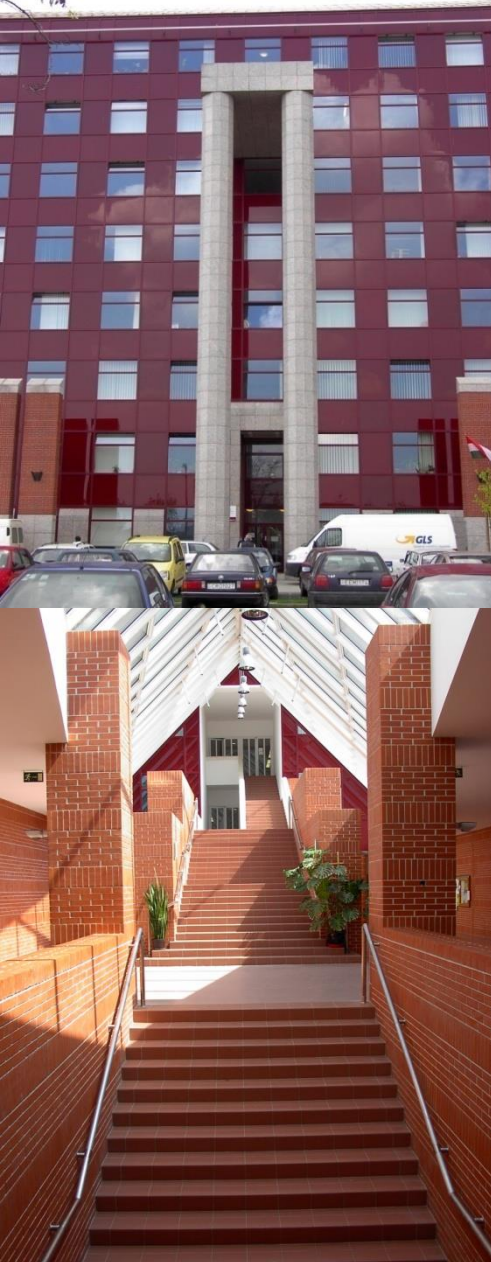
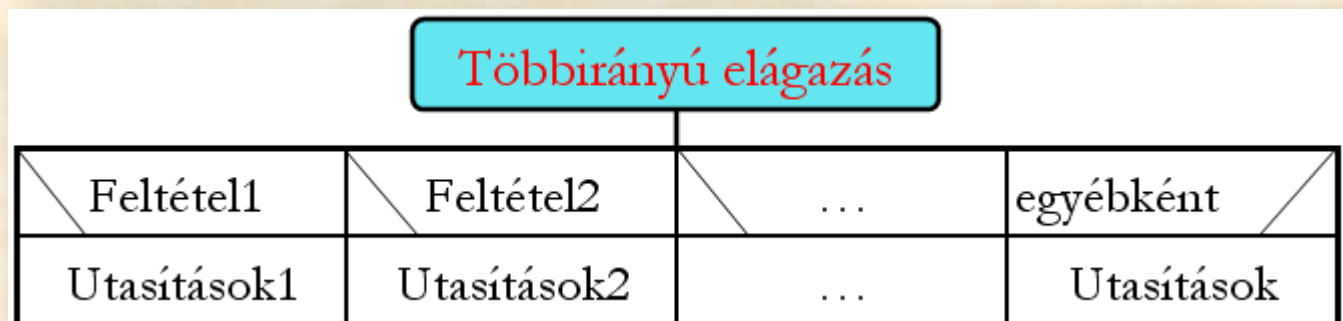
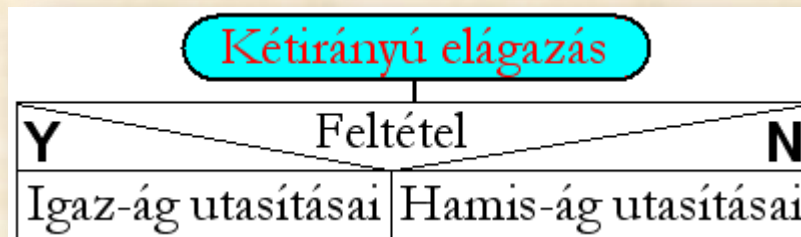
Struktogram (és pseudokód)

➤ Szekvencia:



Utasítás1
Utasítás2

➤ Elágazások:



Struktogram (és pseudokód)

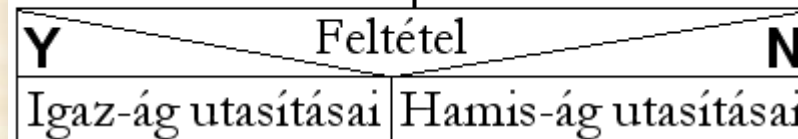
➤ Szekvencia:



Utasítás1
Utasítás2

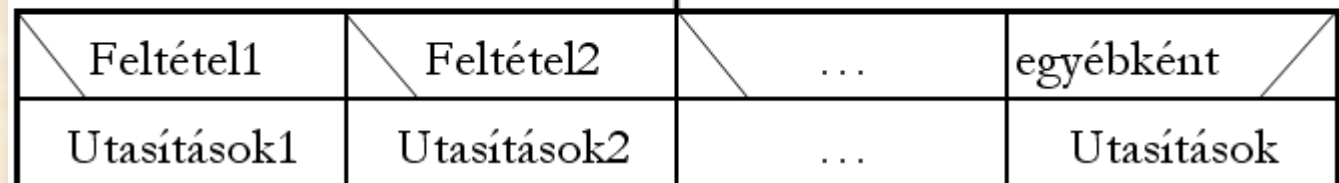
➤ Elágazások:

Kétirányú elágazás



Ha Feltétel **akkor**
Igaz-ág utasításai
különb
Hamis-ág utasításai
Elágazás vége

Többirányú elágazás



Struktogram (és pseudokód)

➤ Szekvencia:

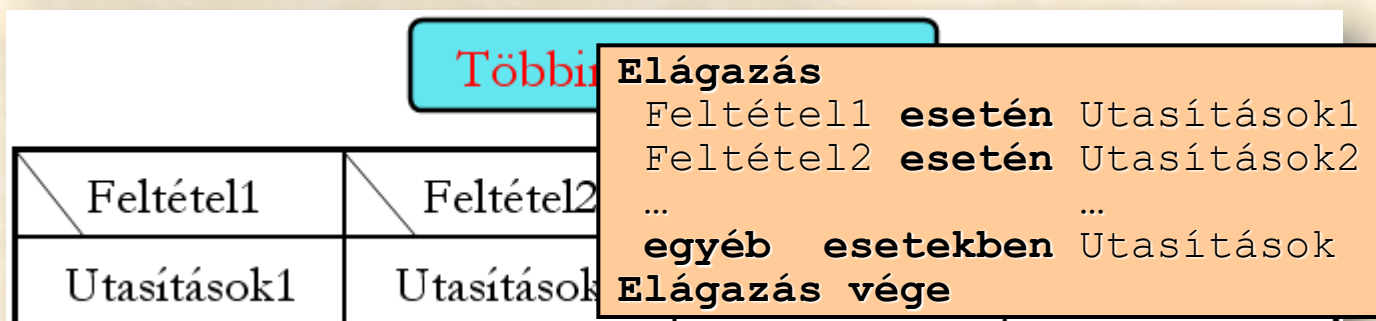


Utasítás1
Utasítás2

➤ Elágazások:



Ha Feltétel **akkor**
Igaz-ág utasításai
különben
Hamis-ág utasításai
Elágazás vége

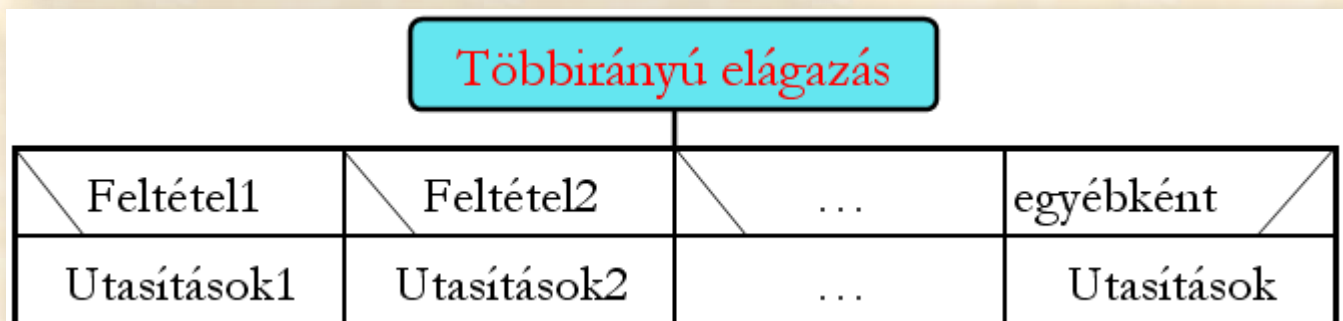


Struktogram (és pseudokód)

➤ Szekvencia:



➤ Elágazások:



Struktogram (és pseudokód)

➤ Ciklusok:

Elöltesztelő ciklus

Bennmaradás feltétele
ciklusmag utasításai

Hátultesztelő ciklus

ciklusmag utasításai
Bennmaradás feltétele

Számlálós ciklus

cv=tól...ig
ciklusmag utasításai



Struktogram (és pseudokód)

➤ Ciklusok:

Elöltesztelő ciklus

Bennmaradás feltétele
ciklusmag utasításai

Ciklus amíg	Feltétel
ciklusmag	utasításai
Ciklus vége	

Hátultesztelő ciklus

ciklusmag utasításai
Bennmaradás feltétele

Számlálós ciklus

cv=tól...ig
ciklusmag utasításai



Struktogram (és pseudokód)

➤ Ciklusok:

Elöltesztelő ciklus

Bennmaradás feltétele
ciklusmag utasításai

Hátultesztelő ciklus

ciklusmag utasításai
Bennmaradás feltétele

Számlálós ciklus

cv=tól...ig
ciklusmag utasításai

Ciklus

ciklusmag utasításai
amíg Feltétel
Ciklus vége



Struktogram (és pseudokód)

➤ Ciklusok:

Elöltesztelő ciklus

Bennmaradás feltétele
ciklusmag utasításai

Hátultesztelő ciklus

ciklusmag utasításai
Bennmaradás feltétele

Számlálós ciklus

cv=tól..ig
ciklusmag utasításai

Ciklus cv=tól ig
ciklusmag utasításai
Ciklus vége



Struktogram (és pseudokód)

➤ Ciklusok:

Elöltesztelő ciklus

Bennmaradás feltétele
ciklusmag utasításai

Hátultesztelő ciklus

ciklusmag utasításai
Bennmaradás feltétele

Számlálós ciklus

$cv = \text{tól} \dots \text{ig}$
ciklusmag utasításai

Számlálós ciklus

$cv = \text{tól}$
 $cv \leq \text{ig}$
ciklusmag utasításai
 $cv = cv + 1$



Struktogram (és pseudokód)

➤ Ciklusok:

Elöltesztelő ciklus

Bennmaradás feltétele
ciklusmag utasításai

Hátultesztelő ciklus

ciklusmag utasításai
Bennmaradás feltétele

Számlálós ciklus

cv=tól...ig
ciklusmag utasításai

➤ Struktogramszerkesztés:

- Táblázatkezelővel/szövegszerkesztővel
- Célprogramokkal (pl. **NSD**)





Kódolás

(fejlesztői környezet)



➤ Keretrendszer:

Code::Blocks

➤ Letöltés:

www.codeblocks.org

➤ Telepítés:

értelemszerűen

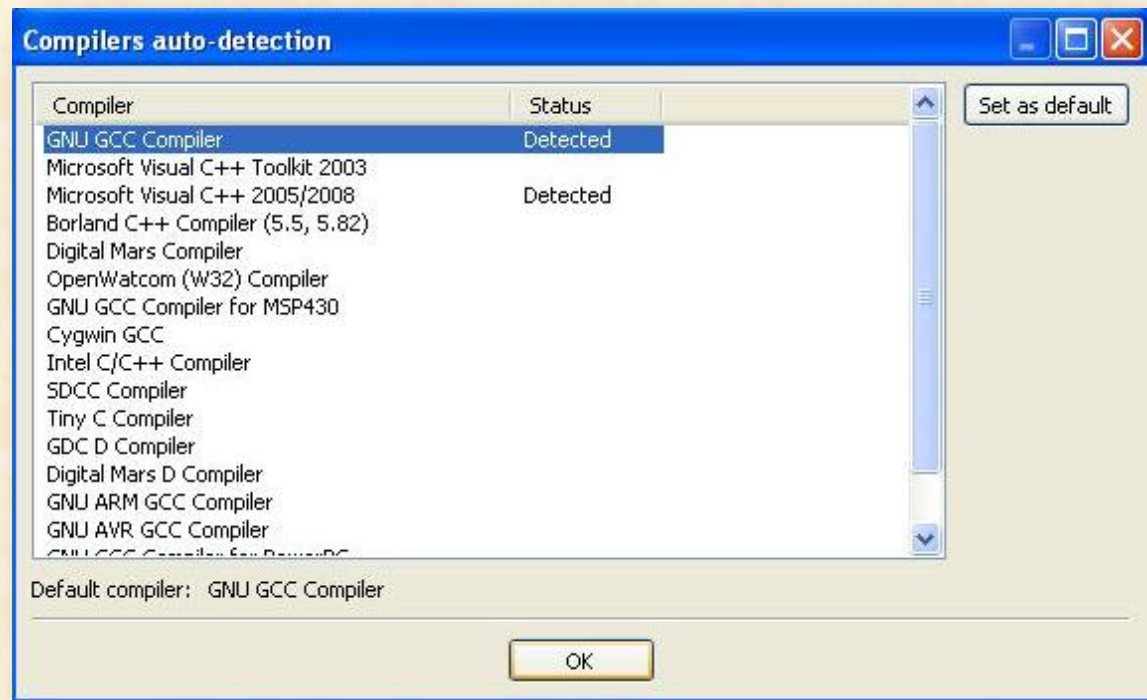


Kódolás

(fejlesztői környezet)



- **Első elindításkor:**
a fordítóprogram kiválasztása



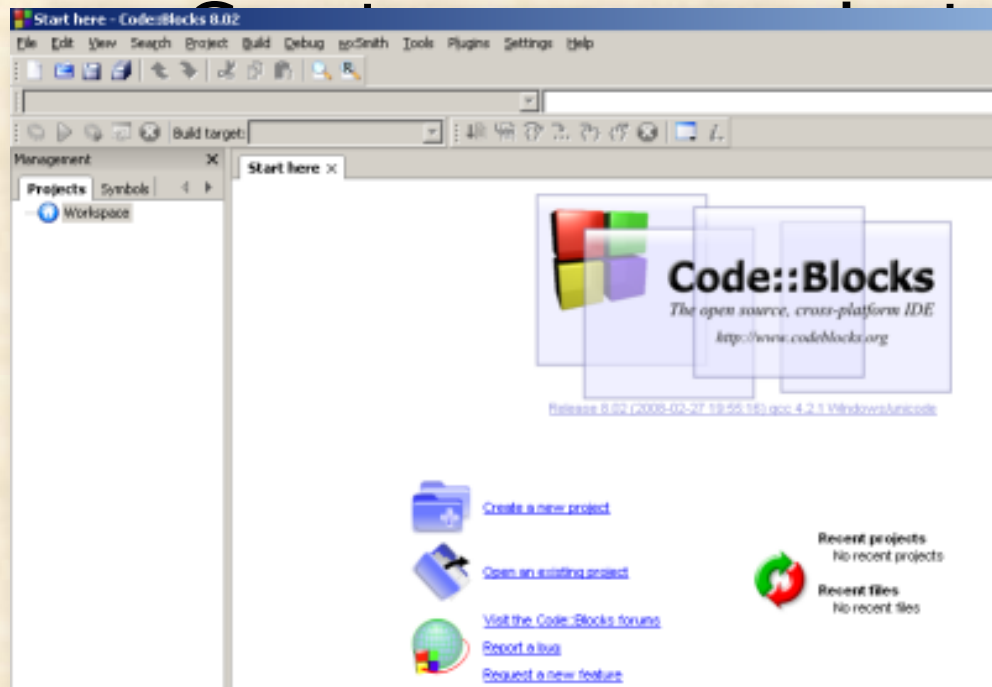
Kódolás

(fejlesztői környezet)



➤ Használat lépései:

1. **projekt** létrehozása, azaz milyen platformra készül a majdani alkalmazás:





Kódolás

(fejlesztői környezet)

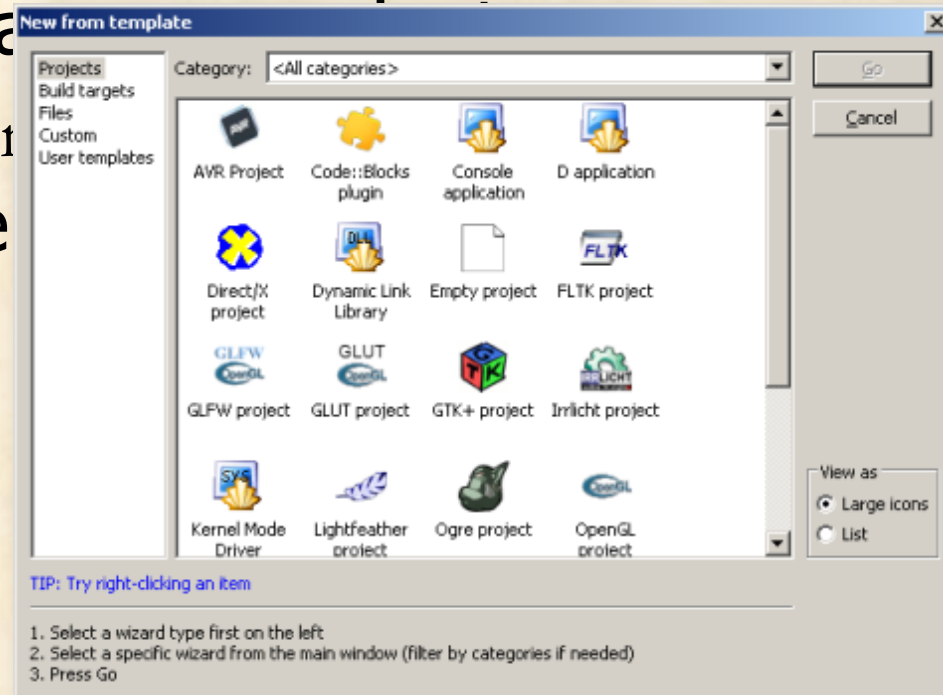


➤ Használat lépései:

1. **projekt** létrehozása, azaz milyen platformra készül a majdani alkalmazás:

Create a

2. sablon (template) kiválasztása:
Console application





Kódolás

(fejlesztői környezet)

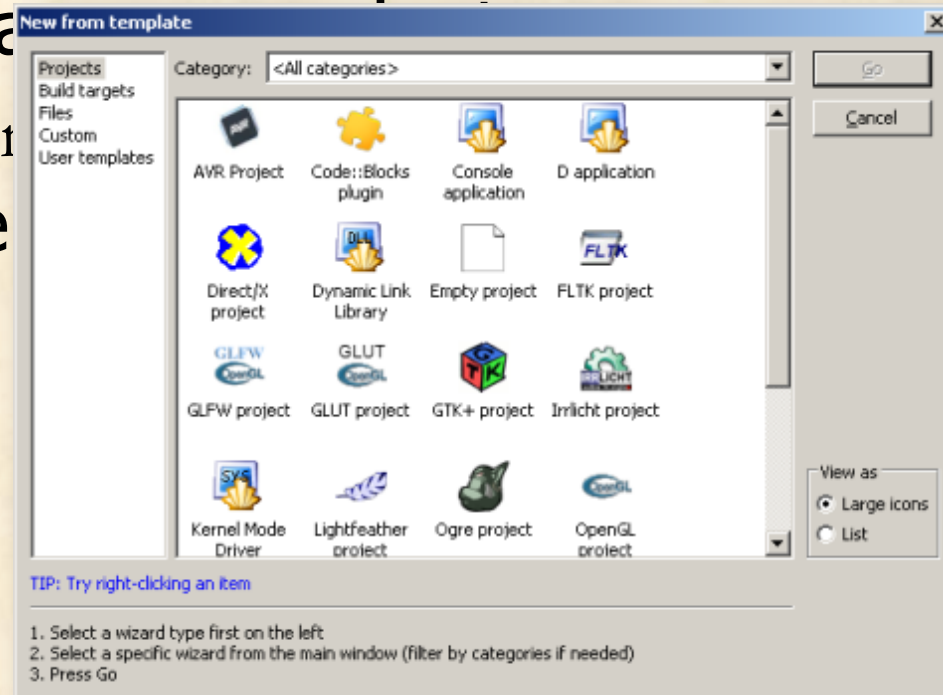


➤ Használat lépései:

1. **projekt** létrehozása, azaz milyen platformra készül a majdani alkalmazás:

Create a

2. sablon (template) kiválasztása:
Console application





Kódolás

(fejlesztői környezet)



➤ Használat lépései:

1. **projekt** létrehozása, azaz milyen platformra készül a majdani alkalmazás:

Create a new project

2. sablon (template) választása:

Console application

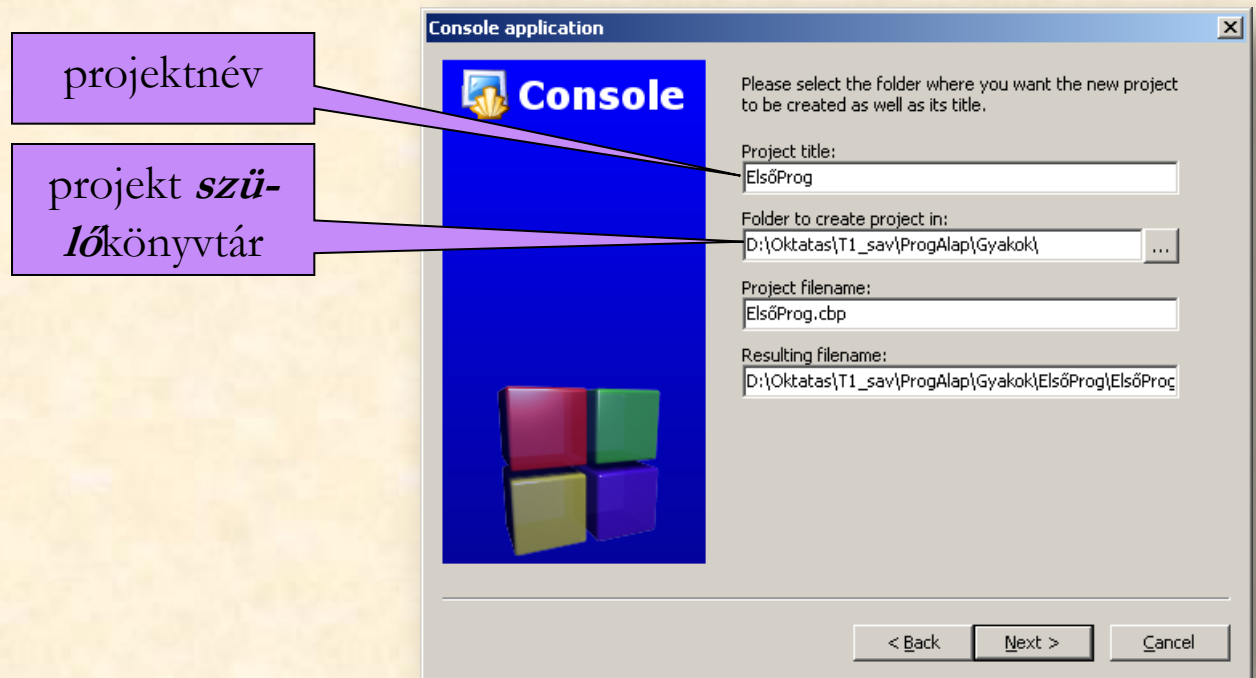


Kódolás

(fejlesztői környezet)



- **Használat további lépései:**
- a projekt munkakörnyezete a diszken





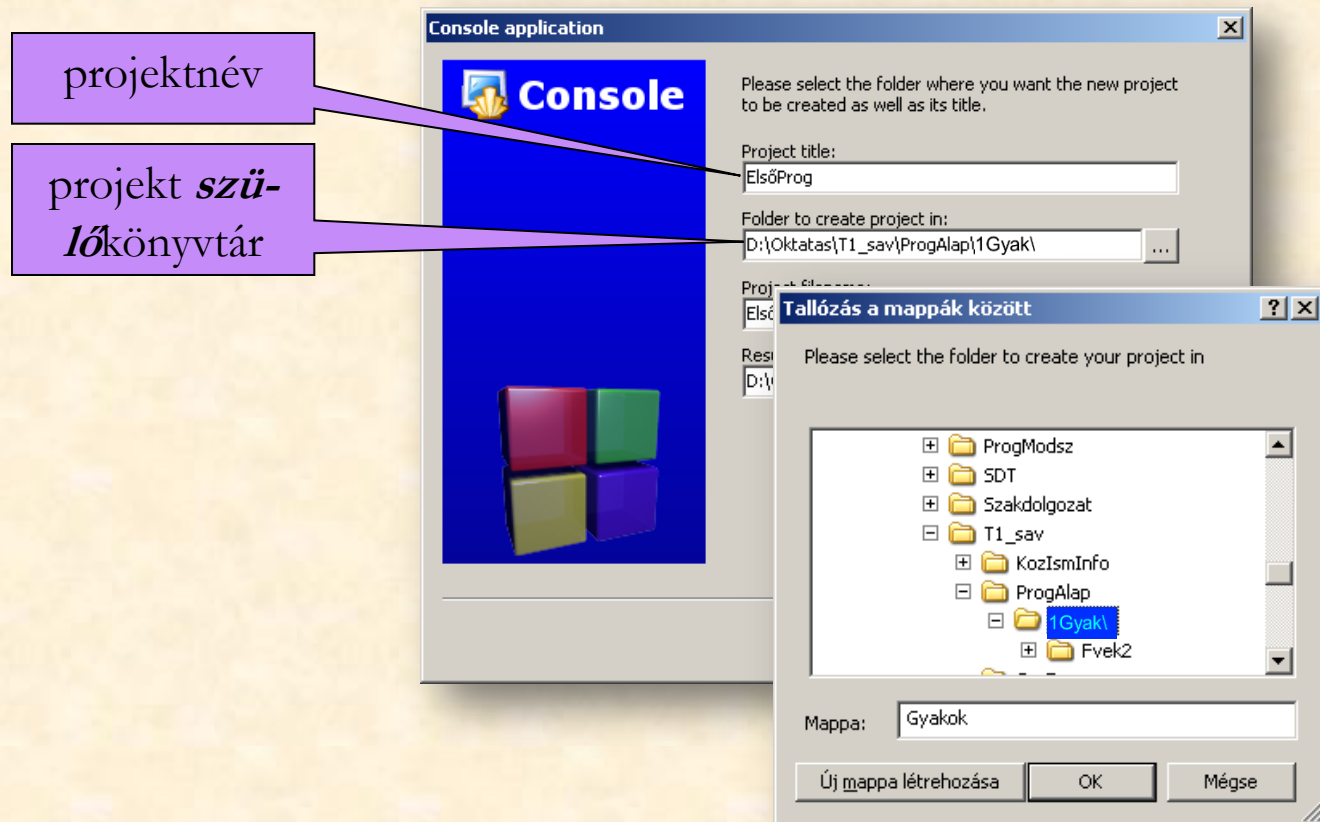
Kódolás

(fejlesztői környezet)



➤ Használat további lépései:

- a projekt munkakörnyezete a diszken





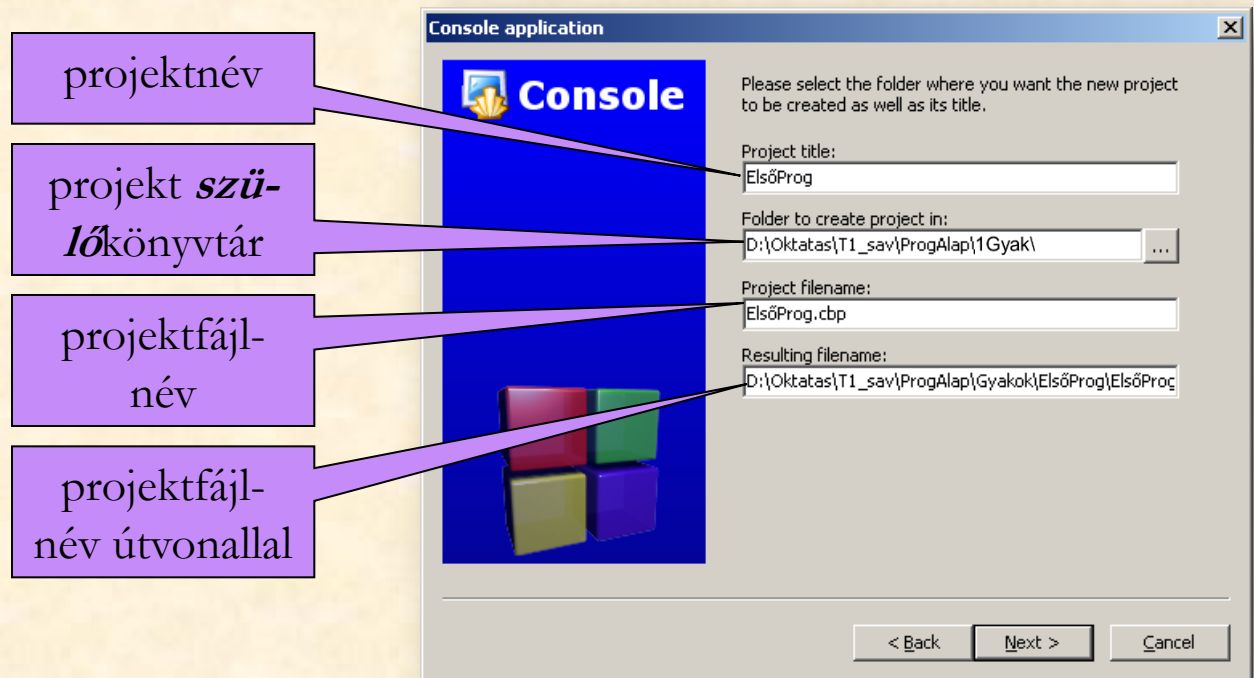
Kódolás

(fejlesztői környezet)



➤ Használat további lépései:

- a projekt munkakörnyezete a diszken





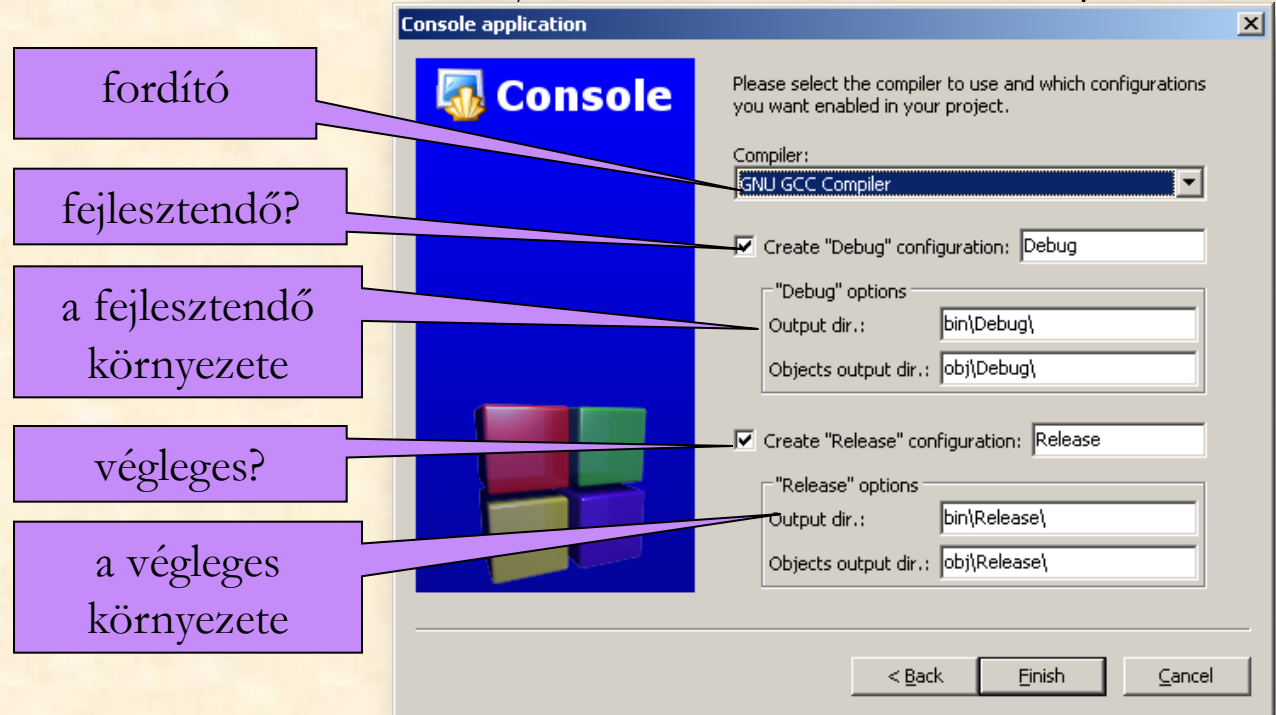
Kódolás

(fejlesztői környezet)



➤ Használat további lépései:

- fordítóválasztás
- a munkakörnyezet kialakítás befejezése





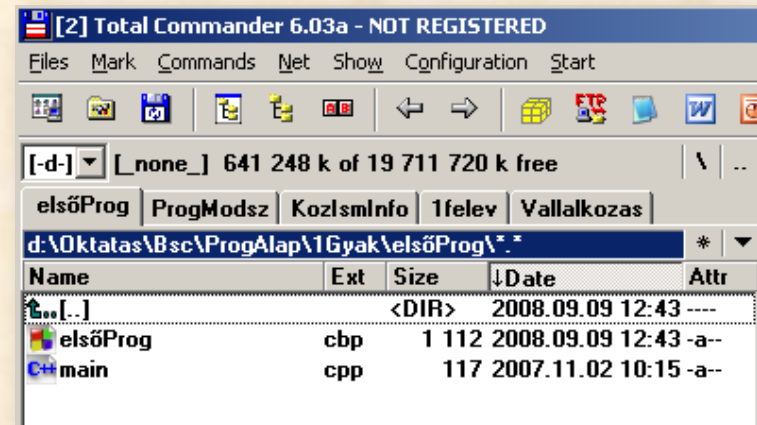
Kódolás

(fejlesztői környezet)

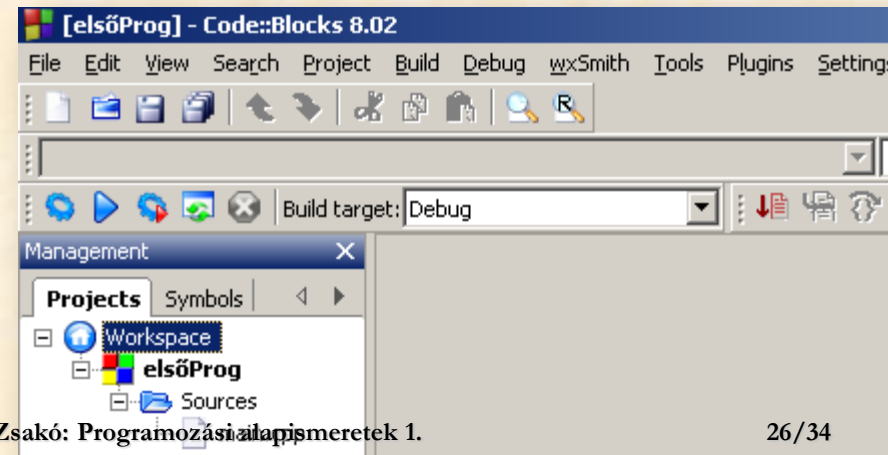


➤ A kialakult munkakörnyezet:

- a diszken:



- a keretrendszerben:





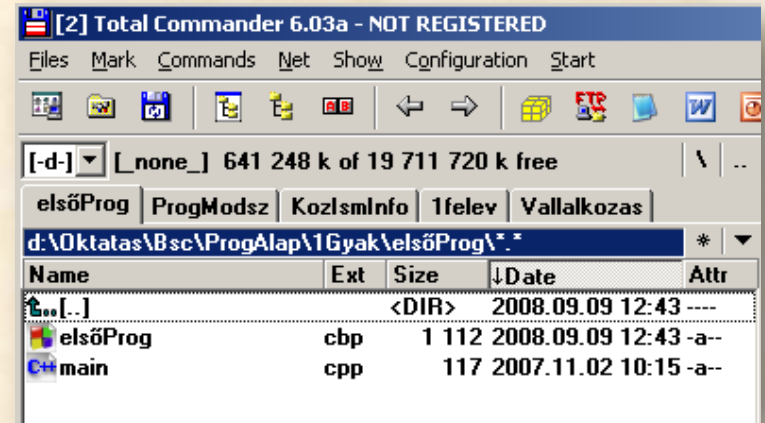
Kódolás

(fejlesztői környezet)



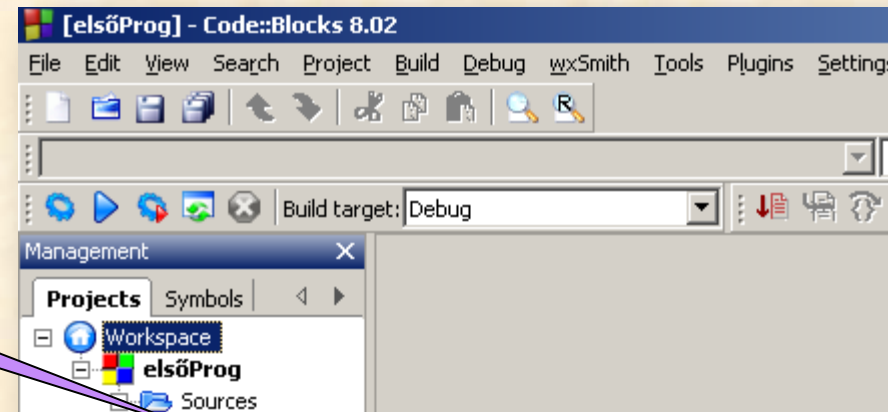
➤ A kialakult munkakörnyezet:

- a diszken:



- a keretrendszerben:

a program
„kibontása”





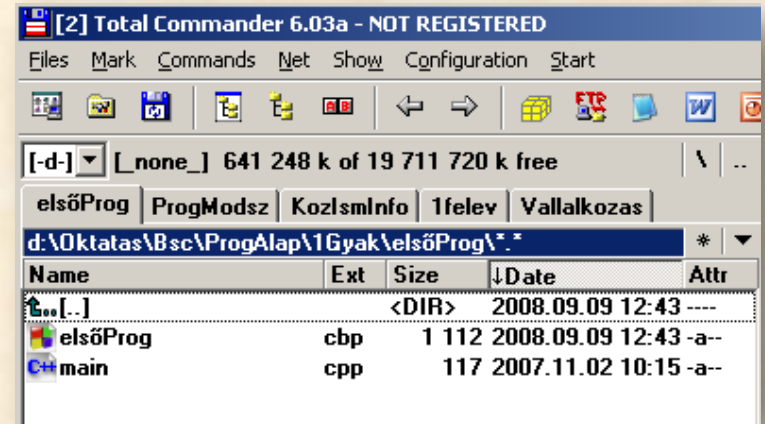
Kódolás

(fejlesztői környezet)



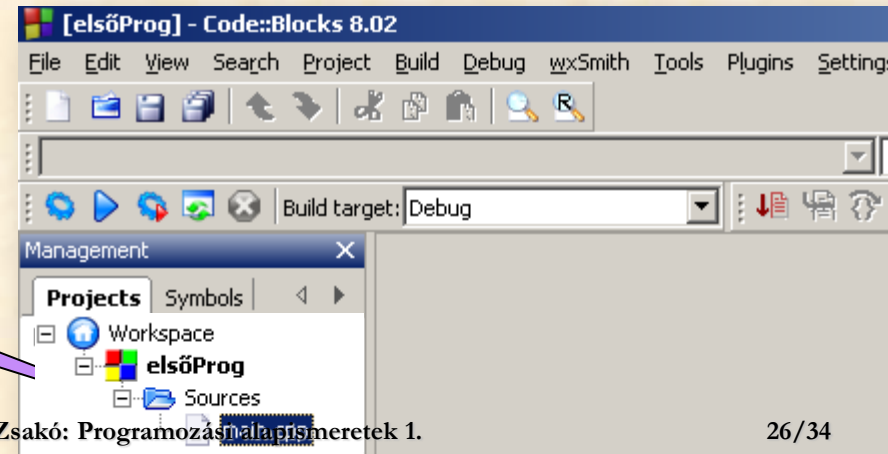
➤ A kialakult munkakörnyezet:

- a diszken:



- a keretrendszerben:

a program
„kibontása”





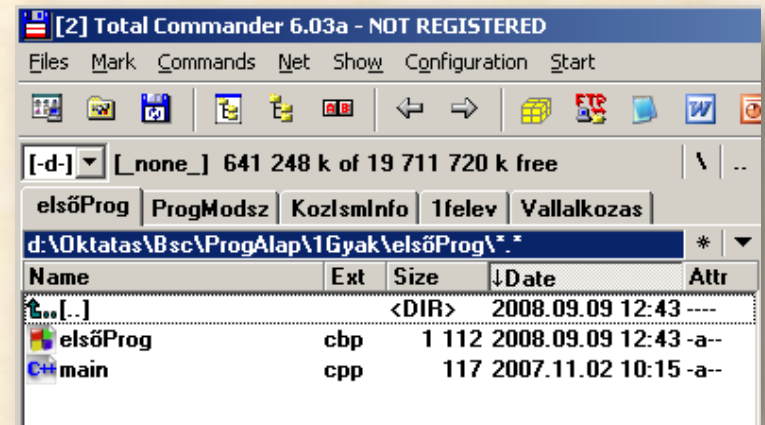
Kódolás

(fejlesztői környezet)

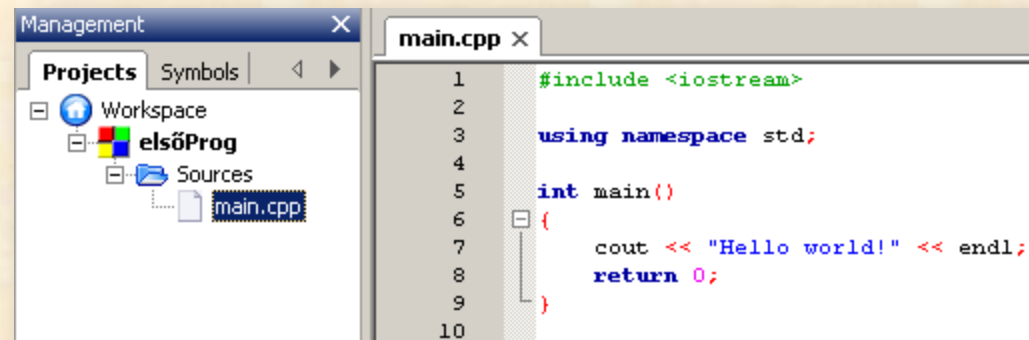


➤ A kialakult munkakörnyezet:

- a diszken:



- a keretrendszerben:



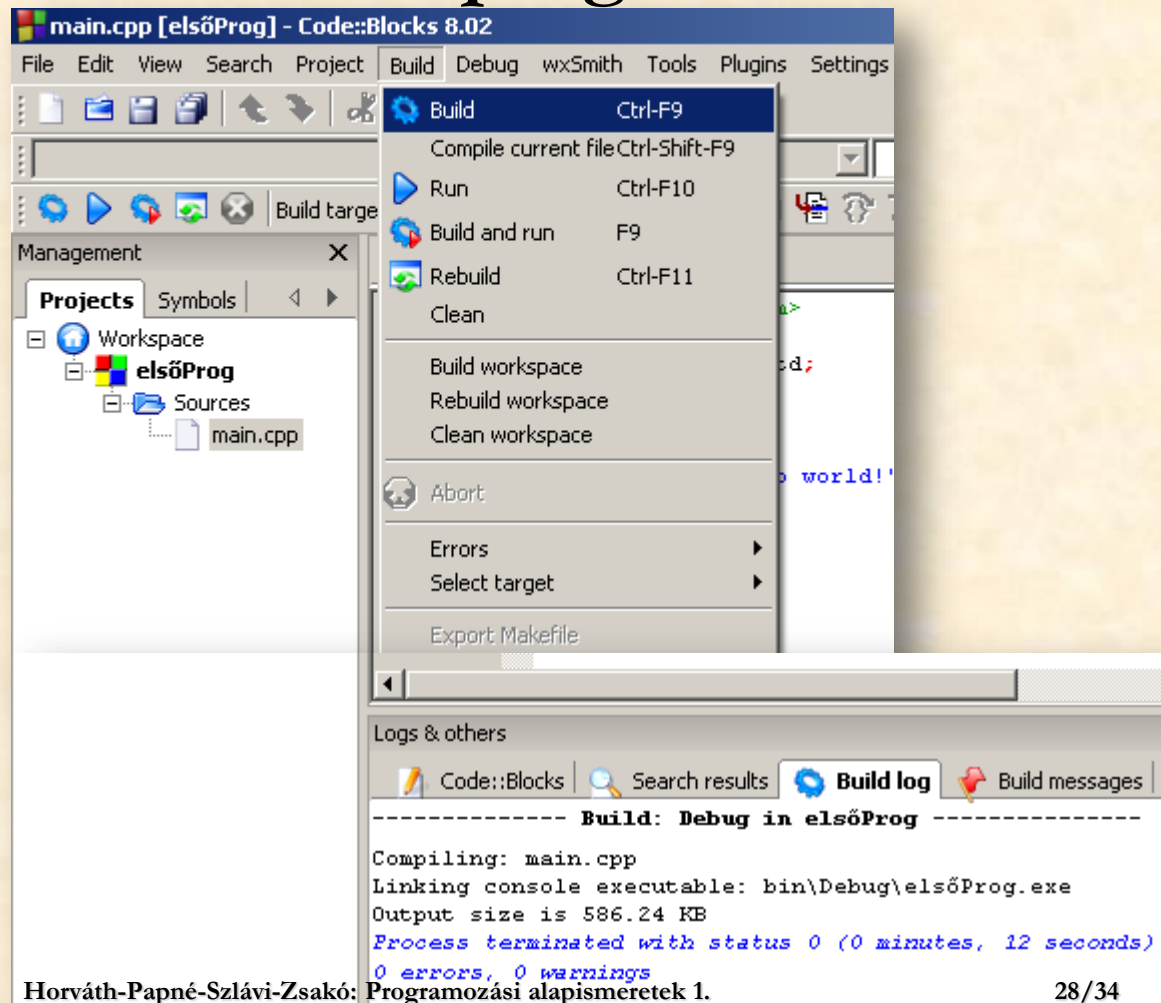


Kódolás

(fejlesztői környezet)



➤ A „keletkezett 0. program” fordítása





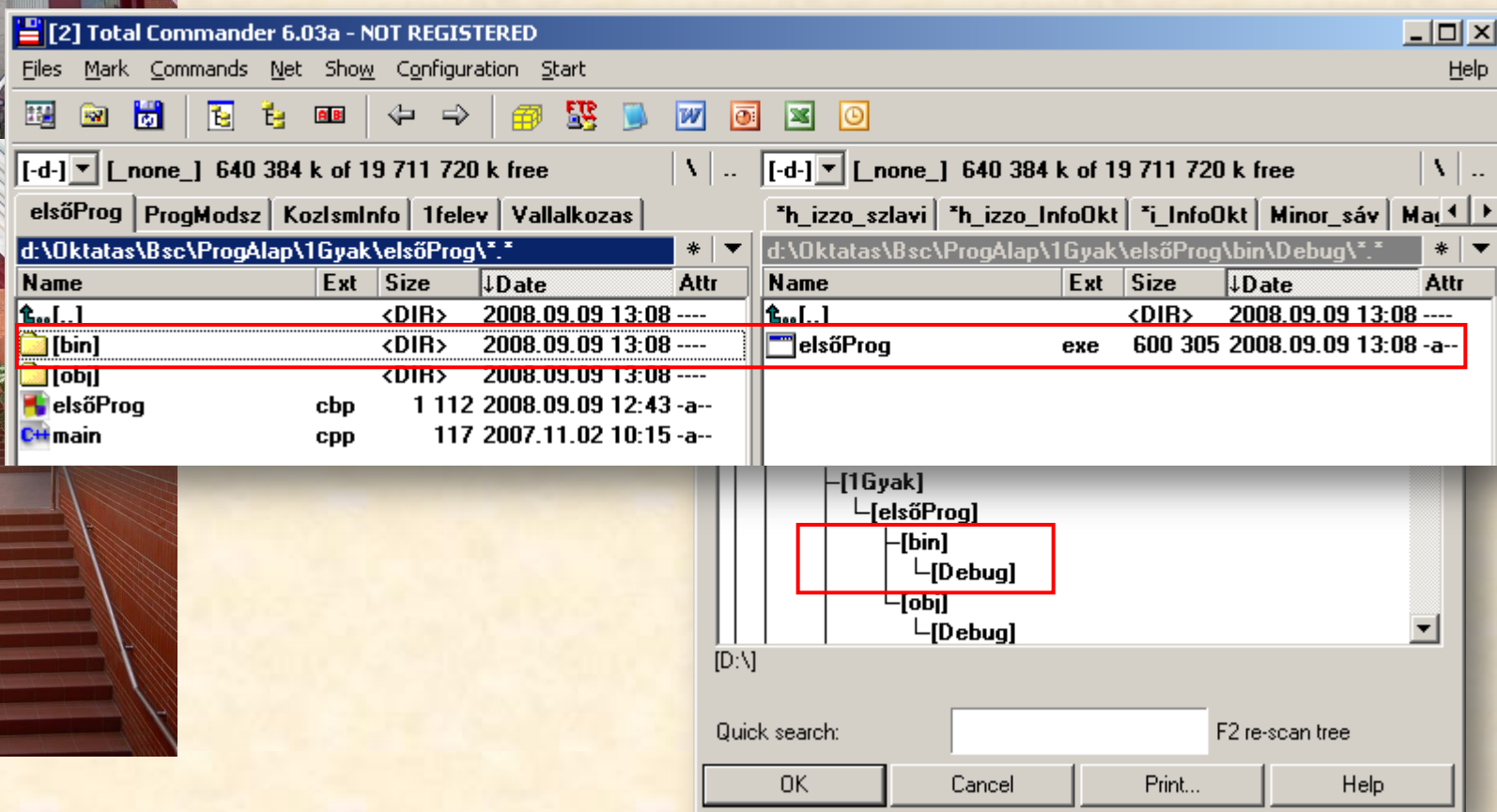
Kódolás

(fejlesztői környezet)



➤ A kialakult munkakörnyezet:

- a diszken:





Kódolás

(fejlesztői környezet)



➤ A kialakult munkakörnyezet:

- a diszken:

[2] Total Commander 6.03a - NOT REGISTERED

Files Mark Commands Net Show Configuration Start Help

[d-] [none_] 640 384 k of 19 711 720 k free

elsőProg ProgModsz KozlsmlInfo 1felev Vallalkozas

d:\Oktatas\Bsc\ProgAlap\1Gyak\elsőProg*

Name	Ext	Size	↓Date	Attr
↑[..]		<DIR>	2008.09.09 13:08	----
[bin]		<DIR>	2008.09.09 13:08	----
[obj]		<DIR>	2008.09.09 13:08	----
elsőProg	cbp	1 112	2008.09.09 12:43	-a--
main	cpp	117	2007.11.02 10:15	-a--

d:\Oktatas\Bsc\ProgAlap\1Gyak\elsőProg\obj\Debug*

Name	Ext	Size	↓Date	Attr
↑[..]		<DIR>	2008.09.09 13:08	----
main	o	114 069	2008.09.09 13:08	-a--

[1Gyak]
└─[elsőProg]
 └─[bin]
 └─[Debug]
 └─[obj]
 └─[Debug]

[D:\]

Quick search: F2 re-scan tree

OK Cancel Print... Help



Kódolás

(fejlesztői környezet)



➤ A kialakult munkakörnyezet:

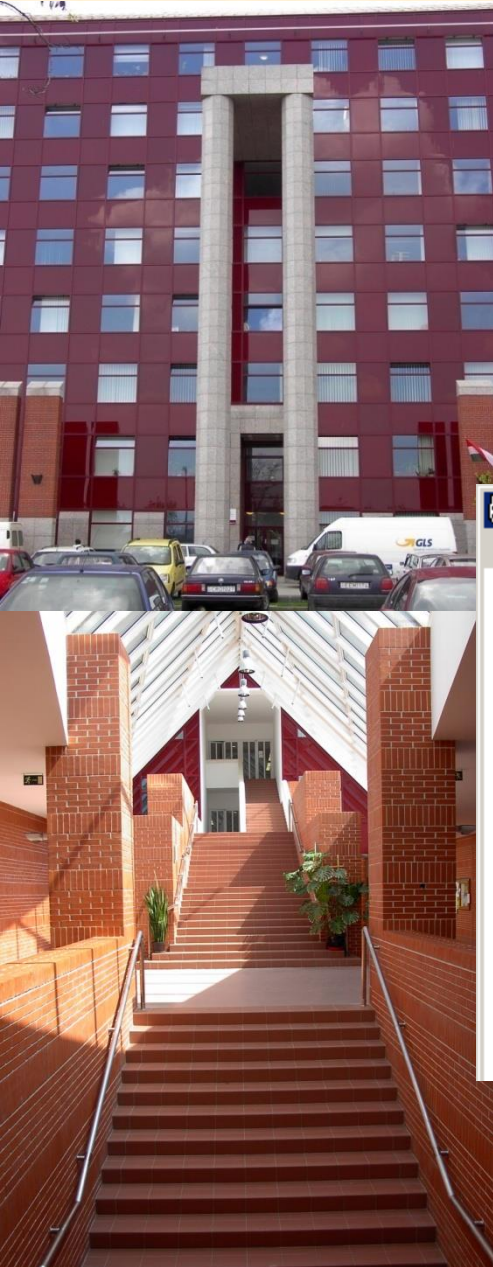
- a **main.cpp** tartalma:

```
#include <iostream>

using namespace std;

int main()
{
    cout << "Hello world!" << endl;
    return 0;
}
```

(meglepő módon?)



Kódolás

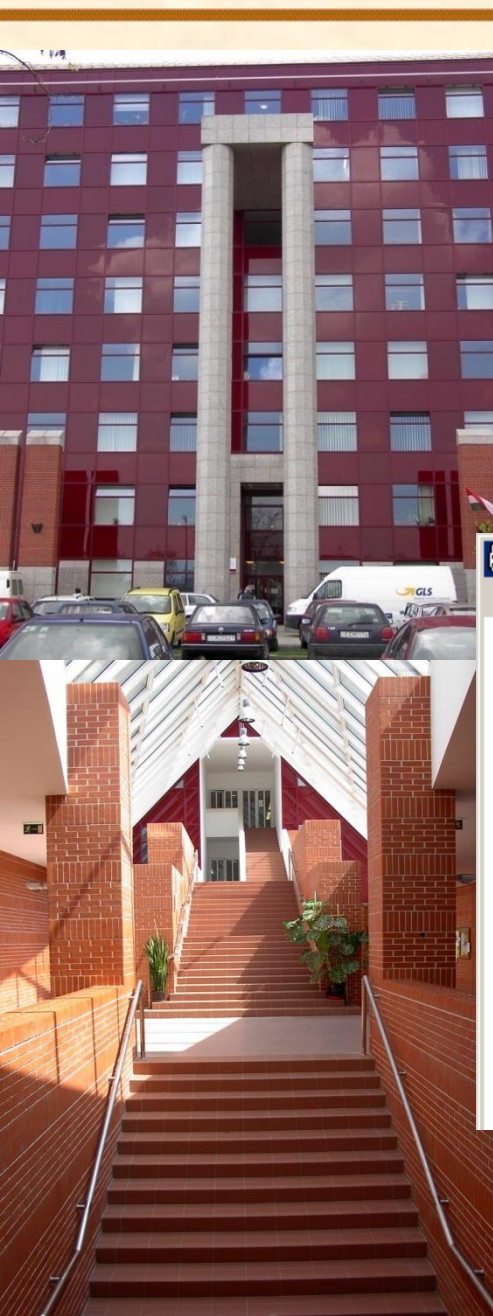
(fejlesztői környezet)



➤ A kialakult munkakörnyezet:

- a **elsőProg.cbp** tartalma (xml):

```
Lister - [D:\Oktatas\Bsc\ProgAlap\1Gyak\elsőProg\elsőProg.cbp]
File Edit Options Help
<?xml version="1.0" encoding="UTF-8" standalone="yes" ?>
<CodeBlocks_project_file>
  <FileVersion major="1" minor="6" />
  <Project>
    <Option title="elsőProg" />
    <Option pch_mode="2" />
    <Option compiler="gcc" />
    <Build>
      <Target title="Debug">
        <Option output="bin\Debug\elsőProg" prefix_auto="1" extension_auto="1" />
        <Option object_output="obj\Debug\" />
        <Option type="1" />
        <Option compiler="gcc" />
        <Compiler>
          <Add option="-g" />
        </Compiler>
      </Target>
```

Kódolás

(fejlesztői környezet)



➤ A kialakult munkakörnyezet:

- a **elsőProg.cbp** tartalma (xml):

```
Lister - [D:\Oktatas\Bsc\ProgAlap\1Gyak\elsőProg\elsőProg.cbp]
File Edit Options Help
<?xml version="1.0" encoding="UTF-8" standalone="yes" ?>
<CodeBlocks_project_file>
<FileVersion major="1" minor="6" />
<Project>
  <Option title="elsőProg" />
  <Option pch <Target title="Release">
  <Option con <Option output="bin\Release\elsőProg" prefix_auto="1" extension_auto="
  <Build> <Option object_output="obj\Release\" />
    <Target ti <Option type="1" />
      <Option c <Option compiler="gcc" />
        <Option c <Compiler>
          <Option t <Add option="--02" />
            <Option c </Compiler>
              <Compiler <Linker>
                <Add opt <Add option="-s" />
                  </Compile </Linker>
                    </Target>
  </Build>
  <Compiler>
    <Add option="--Wall" />
    <Add option="--fexceptions" />
  </Compiler>
  <Unit filename="main.cpp" />
  <Extensions>
```



Kódolás

(fejlesztői környezet)



➤ A kialakult munkakörnyezet:

- a **elsőProg.cbp** tartalma (xml):



(mily meglepő!)



Kódolás

(fejlesztői környezet)



➤ A futó konzolalkalmazás:

- „fordítás” –
- (az utolsó lefordított) futtatás(a) –
- fordítás+futtatás –
- és a konzolablak tartalma:

```
C:\ D:\Oktatas\Bsc\ProgAlap\1Gyak\els$Prog\bin\Debug\els$Prog.exe
Hello world!

Process returned 0 (0x0)   execution time : 0.469 s
Press any key to continue.
```



Kódolás

(fejlesztői környezet)



➤ A futó konzolalkalmazás:

- „fordítás” –
- (az utolsó lefordított) futtatás(a) –
- fordítás+futtatás –
- és a konzolablak tartalma:

a futás
eredménye

```
C:\ D:\Oktatas\Bsc\ProgAlap\1Gyak\els$Prog\bin\Debug\els$Prog.exe
Hello world!

Process returned 0 (0x0)   execution time : 0.469 s
Press any key to continue.
```


Kódolás

(fejlesztői környezet)



➤ A futó konzolalkalmazás:

- „fordítás” –
- (az utolsó lefordított) futtatás(a) –
- fordítás+futtatás –
- és a konzolablak tartalma:

```
C:\ D:\Oktatas\Bsc\ProgAlap\1Gyak\els$Prog\bin\Debug\els$Prog.exe
Hello world!

Process returned 0 (0x0)   execution time : 0.469 s
Press any key to continue.
```

a futás
eredménye

visszaadott
érték

Kódolás

(fejlesztői környezet)



➤ A futó konzolalkalmazás:

- „fordítás” –
- (az utolsó lefordított) futtatás(a) –
- fordítás+futtatás –
- és a konzolablak tartalma:

futási idő

a futás
eredménye

visszaadott
érték

```
C:\D:\Oktatas\Bsc\ProgAlap\1Gyak\elsProg\bin\Debug\elsProg.exe
Hello world!
Process returned 0 (0x0)   execution time : 0.469 s
Press any key to continue.
```


Kódolás

(fejlesztői környezet)



➤ A futó konzolalkalmazás:

- „fordítás” –
- (az utolsó lefordított) futtatás(a) –
- fordítás+futtatás –
- és a konzolablak tartalma:

futási idő

a futás
eredménye

visszaadott
érték

```
C:\D:\Oktatas\Bsc\ProgAlap\1Gyak\els$Prog\bin\Debug\els$Prog.exe
Hello world!
Process returned 0 (0x0)   execution time : 0.469 s
Press any key to continue.
```

Érdeemes elindítani az exe-t!
Mit tapasztalt? Magyarázat?



The background of the slide is a black and white aerial photograph of Budapest, Hungary. The Danube River flows through the city, with the Buda side visible on the left and the Pest side on the right. The city skyline is dense with buildings, and the Parliament Building is prominent in the foreground. A semi-transparent white rectangle is overlaid on the center of the image, containing the text.

Programozási alapismeretek 1. előadás vége