

```
/* 2. felev, uj tematika, C++ nyelv      A C++ nem obj,orient.ujdonsagai
   8. gyak      GEPESZ                  Egyszeru pelda osztalyra es
                                           egyedre.
```

```
// 1. prg: cimszerinti ertekatadas referenciavaltozok segitsegevel:
```

```
/*
#include <stdio.h>
#include <conio.h>
#include <iostream.h> //ez az osztalydeklaracio kell a cin es cout-hoz

void szelsoertekek(int a, int b, int& max, int& min)
{
    if (a > b) { max= a; min= b;} else {max= b; min= a;};
}

void main(void)
{ int c, d, nagy, kicsi;
  clrscr();
  cout << "Cimszerinti ertekatadas demo" << endl;
  do
  {cout << "Adj meg ket egesz szamot!" << endl;
   cout << "c=";
   cin >> c;
   cout << "d=";
   cin >> d;
   szelsoertekek(c, d, nagy, kicsi);
   cout << "A nagyobbik= " << nagy << ", a kisebbik= " << kicsi << endl;
  }
  while (c!=d);
  cout << "vege." ;
  getch();
}

*/
```

```
//2. pelda: Dinamikus memoriakezeles new es delete segitsegevel:
```

```
/*
#include <stdio.h>
#include <conio.h>
#include <stdlib.h>
#include <iostream.h>

//probaljuk ki nagyobb ertekkel is, pl 1600 !
#define db 16

void main()
{
```

```
randomize();
float * valosmut;
float * vektmut;
clrscr();
cout << "Dinamikusan foglalunk helyet egy valos skalarnak \
      es egy vektornak" << endl;
valosmut = new float;
vektmut  = new float[db];
if (!valosmut || !vektmut)
  { cout << "Memoriafoglalas sikertelen!" << endl; exit(1); }
*valosmut= random(25)/11.5;
for (int i=0; i<db; i++) vektmut[i]= random(25)/12.0;
//kiiratas:
for (i=0; i<db; i++)
  cout << "vektmut[" << i << "]= " << vektmut[i] << endl;
cout << "*valosmut= " << *valosmut << endl;
cout << "Majd felszabaditjuk a foglalt helyeket. " << endl;
delete valosmut;
delete vektmut; //delete[] vektmut; Borland C++ Builder eseteben
getch();
}
*/
```

//3. pelda: Alapertelmezett fuggvenyargumentum, osztaly, egyed pelda
// Lásd előadásanyagot is!

```
#include <stdio.h>
#include <conio.h>
#include <string.h>
#include <iostream.h>
#include <ctype.h>

class Szovegparos
{private: char forras[80],
          cel[80];
public: void MyStrncpy(int honnan=0, int mennyit=2000);
        void SetForras(char forrasbe[]){strcpy(forras,forrasbe);}
        void PrintCel(){puts(cel);}
};

Szovegparos obj;
char forrasbe[80];
int honnan, mennyit;
char c1,c2;

void main()
{clrscr(); c1='N'; c2='N';
  puts("Add meg az s1 szoveget:");
  gets(forrasbe); obj.SetForras(forrasbe);
  cout << "Megadja a honnan poziciot? (I/N): " << endl;
```

```
if ((c1=toupper(getch())) == 'I')
    {cout << "honnan= " ; cin >> honnan;}
if (c1=='I')
{
    cout << "Megadja a mennyit erteket? (I/N): " << endl;
    if ((c2=toupper(getch())) == 'I')
        {cout << "mennyit= " ; cin >> mennyit;}
}
if (c2=='I')    obj.MyStrncpy(honnan, mennyit);
else if (c1=='I') obj.MyStrncpy(honnan);
                else    obj.MyStrncpy();
obj.PrintCel();
getch();
}

void Szovegparos::MyStrncpy(int honnan, int mennyit)
{if (mennyit==2000) strncpy(cel, forras+honnan, strlen(forras)-honnan+1);
 else strncpy(cel, forras+honnan, mennyit);
}
//forras: kerekparverseny, honnan: N, kiir: kerekparverseny
//forras: kerekparverseny, honnan=8, meddig: N : kiir:verseny
//forras: kerekparverseny, honnan=5, meddig: 3 : kiir: par
```

From:

<https://edu.iit.uni-miskolc.hu/> - Institute of Information Science - University of Miskolc

Permanent link:

https://edu.iit.uni-miskolc.hu/tanszek:oktatas:c_programozas?rev=1303211952

Last update: **2011/04/19 11:19**

