

Programozás I.

Szekeres György

Eddig tanultuk

- Kiírás képernyőre
- Ékezetes karakterek kezelése
- Változók használata
- Matematikai műveletek
- Beolvasás konzolról
- Float, double, char
- If-else elágazás
- For ciklus
- While, do-while ciklusok
- Switch-case elágazás
- Tömbök
- Karakterkezelő függvények
- Többdimenziós tömbök
- Struktúrák
- String típus
- String osztály tagfüggvények
- Kiíratás
- Véletlenszám generálása
- Fájlkezelés
- Tagfüggvények

Mutató

- olyan változók, amelyek más változók **címét** tartalmazzák
- nem csak egyetlen változó címét tárolhatja, hanem egy tömb celláinak címét is
- definiálás: típus *azonosító; pl.: int *mut;
- értékadás: mut=&tmb[i]; pl.: mut=&tmb[0]; // tmb első elemére mutat
 (mut=tmb;)

Mutató példa

```
#include <iostream>
using namespace std;
int main()
{
    float tmb[3];
    float* mut;

    cout << "Memoriacimek megjelenítése tömb segítségével: " << endl;
    for (int i = 0; i < 3; ++i)
    {
```

```
        cout << "&tmb[" << i << "] = " << &tmb[i] << endl;
    }
    mut = tmb; // mut = &tmb[0] mutató értéke a tömb első
               // eleme
    cout << "Memoriacimek megjelenítése mutató segítségével: "
         << endl;
    for (int i = 0; i < 3; ++i)
    {
        cout << "mut + " << i << " = " << mut + i << endl;
    }
    return 0;
}
```

Feladat

- Egy munkahely dolgozóinak (maximum 20 fő) május havi adatait tároljuk (majus.txt), a nevük, a születési évük, a ledolgozott órák száma és az aktuális fizetés. A dolgozók idénymunkások, így nem mindenki dolgozik minden hónapban és nettó 2800Ft-os óradíjban dolgoznak. Végezze el a következő feladatokat:
 1. Hozzunk létre egy megfelelő adatszerkezetet az adatok tárolására
 2. Írjuk ki hány fő dolgozott májusban
 3. Töltsük fel az adatszerkezetet adatokkal úgy, hogy a nevet fájlból olvassuk be, a születési évet (1957-2004 lehet!) ellenőrzötten kérjük be a billentyűzetről adott névhez, a ledolgozott órák számát (0-160 lehet) véletlengenerátorral állítsuk elő, a fizetést pedig számoljuk ki
 4. Írassuk ki az adatokat táblázatosan
 5. Számítsuk ki az összes ledolgozott órák számát
 6. Számoljuk meg hányan dolgoztak májusban 100 óránál többet
 7. Írassuk ki a dolgozók átlag életkorát
 8. Írjuk ki az utolsóként tárolt dolgozó nevét, ledolgozott óráinak számát és a havi bruttó bérét
 9. Mentsük a teljes adatokat **majus_teljes.txt** néven

Feladat

```
#include <iostream>
#include<string>
#include<ctime>
#include<fstream>
#include<iomanip>
using namespace std;
struct dolgozo
{
    string nev;
    int ev, oraszam, fizetes;
};
int main()
{
```

```
    dolgozo majus[20];
    int i, szaz = 0, ossz = 0, eletkor = 0, letszam=0;
    srand((unsigned)time(NULL));
    ifstream jelentes("majus.txt");
    for (i = 0; i < 20 && !jelentis.eof(); i++) // adatok feltöltése
    {
        if (jelentis.is_open()) jelentis >> majus[i].nev;
        else "A fájl megnyitása sikertelen.";
        do
        {
            cout << "\nAdja meg "<<majus[i].nev<<" dolgozo születési
            evet: ";
            cin >> majus[i].ev;
            cin.ignore();
```

Feladat

```
        if (majus[i].ev < 1957 || majus[i].ev > 2004) cout << "Hibas adat,  
a szulesi ev 1957-2004 kozott lehet!" << endl;  
    } while (majus[i].ev < 1957 || majus[i].ev > 2004);  
    majus[i].oraszam = rand() % 160 + 1;  
    majus[i].fizetes = majus[i].oraszam * 2800;  
    letszam++;  
}  
jelentes.close();  
system("cls");  
cout << "Osszesen " << letszam << " fo dolgozott majusban." <<  
endl;
```

```
cout.setf(ios::left); // 4. - eredmények kiíratása "táblázatosan"  
for (i = 0; i < letszam; i++)  
{  
    cout << setw(20) << majus[i].nev << majus[i].ev << "\t\t" <<  
    majus[i].oraszam << " ora\t\t" << majus[i].fizetes << " Ft" << endl;  
    ossz = ossz + majus[i].oraszam; // 5. - összes óraszám  
    if (majus[i].oraszam > 100)  
        szaz++; // 6. - 100 óránál több  
    eletkor = eletkor + (2022 - majus[i].ev); // 7. - átlagéletkor  
}
```

Feladat

```
cout << "\nMajus havi osszes ledolgozott orak szama: " << ossz << endl;

cout << szaz << " dolgozo teljesített 100 oranal tobbet majus honapban." << endl;

cout << "\nA majusban dolgozok atlag eletkora: " << (float)eletkor / letszam << " ev." << endl;

// h. feladat - elsőként tárolt dolgozó adatai

cout << "\nAz utolsokent rogzített dolgozo adatai: \n\t" << majus[letszam - 1].nev << "\t születési éve: " << majus[letszam - 1].ev << "\t majusi oraszama: " << majus[letszam - 1].oraszam << "\t brutto fizetese: " << majus[letszam - 1].fizetes * 1.38 << " Ft" << endl;

ofstream ksz("majus_teljes.txt");

if (ksz.is_open())

{
```

```
    for (i = 0; i < letszam; i++)
    {
        ksz << majus[i].nev << "\t";
        ksz << majus[i].ev << "\t";
        ksz << majus[i].oraszam << "\t";
        ksz << majus[i].fizetes << endl;
    }
    ksz.close();

    cout << "\nA lista sikeresen elkeszult." << endl;
}

else

    cout << "\nA lista létrehozása sikertelen." << endl;

    return 0;

}
```