

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

Feladat

- Egy vegyesbolt adatait kell tárolni. Tároljuk az áruk neveit, árát, és a raktáron lévő darabszámot! Hozzunk létre egy az előbbiek tárolására szolgáló struktúra változót! Maximum 15db különféle árunk lehet. Végezze el az alábbi feladatokat:
 1. Kérjük be az áruk számát ellenőrzötten!
 2. Kérjük be az áruk neveit! (pl: Fekete ribizli szörp)
 3. Kérjük be az áruk árát ellenőrzötten (100 és 5000Ft között lehet)
 4. Találja ki a darabszámokat véletlengenerátorral (max. 10 db lehet)
 5. Írja ki az adatokat táblázatosan
 6. Határozza meg, és írja ki, mennyi az egész árukészlet értéke
 7. Számolja ki a termékek átlagos árát és írja ki a képernyőre
 8. Számolja meg hány áruból van több mint 5 raktárkészleten
 9. Írja ki a legdrágább áru nevét és árát
 10. Írja ki az adatokat egy aktualis.txt nevű fájlba tabulátorral tagoltan.
 11. Emelje ki az árakat 20%-kal majd írja ki az áruk nevét és az új árát.

Feladat

```
#include <iostream>
#include<iomanip>
#include<string>
#include<ctime>
#include<fstream>
using namespace std;
struct Aru
{
    string arunev;
    int ar, db;
};
int main()
{
    Aru bolt[15];
```

```
    int i, legdragabb, ertek = 0, atlagar = 0, aruszam;
    srand((unsigned)time(NULL));
    // adatbekérés - különböző áruk darabszámának bekérése ellenőrzötten
    do
    {
        cout << "Adja meg hany fele aru van a boltban ! (1-15)" << endl;
        cin >> aruszam;
        cin.ignore();
        if (arszam < 1 || arszam>15) cout << "Hibas adat, az aruk szama 1-15 lehet!" << endl;
    } while (arszam < 1 || arszam>15);
```

Feladat

// adatbekérés - különböző áruk neve, darabszáma és ára

```
for (i = 0; i < aruszam; i++)
{
    cout << "Adja meg az " << i + 1 << " . aru nevet\n";
    getline(cin, bolt[i].arunev);

do
{
    cout << "Adja meg az aru arat (100-5000Ft)\n";
    cin >> bolt[i].ar;
    cin.ignore();
    if (bolt[i].ar < 100 || bolt[i].ar > 5000)
        cout << "Hibas adat, az aruk ara 100-5000Ft lehet!" << endl;
} while (bolt[i].ar < 100 || bolt[i].ar > 5000);
bolt[i].db = rand() % 10 + 1;
}
```

// 5. feladat eredmények kiírása "táblázatosan"

```
system("cls");
cout.setf(ios::left);
for (i = 0; i < aruszam; i++)
{
    cout << setw(20) << bolt[i].arunev << bolt[i].ar << "Ft" << "\t" <<
    bolt[i].db << "db" << endl;
}
```

// 6. feladat - egész árukészlet értéke

```
for (i = 0; i < aruszam; i++)
{
    ertek = ertek + bolt[i].ar * bolt[i].db;
}

cout << "\nAz arukeszlet ossz ereteke: " << ertek << "Ft" << endl;
```


Feladat

// 7. feladat - átlagár kiszámítása

```
for (i = 0; i < aruszam; i++)
{
    atlagar += bolt[i].ar;
}

cout << "\nAz aruk atlagara: " << atlagar/aruszam << "Ft" << endl; //
(nincs szükség a törtrészre)
```

// 8. feladat - hány áruból van több mint öt raktárkészleten

```
int otneltobb = 0;

for (i = 0; i < aruszam; i++)
{
    if (bolt[i].db > 5) otneltobb++;
}

cout << endl << otneltobb << " arubol van otnel tobb a keszleten."
<< endl;
```

//9. feladat - Írja ki a legdrágább áru nevét és árát!

```
legdragabb = 0;

for (i = 1; i < aruszam; i++)
{
    if (bolt[i].ar > bolt[legdragabb].ar) legdragabb = i;
}

cout << "\nA legdragabb aru adatai: \n" << setw(20) <<
bolt[legdragabb].arunev << "\t" << bolt[legdragabb].ar << "Ft" <<
endl;
```

Feladat

//10. feladat Lista kiírása fájlba

```
ofstream keszlet("aktualis.txt");
if (keszlet.is_open())
{
    for (i = 0; i < aruszam; i++)
    {
        keszlet << bolt[i].arunev << "\\t";
        keszlet << bolt[i].ar << "\\t";
        keszlet << bolt[i].db << endl;
    }
    keszlet.close();

    cout << "\\nA lista sikeresen elkészült." << endl;
}
else cout << "\\nA lista létrehozása sikertelen." << endl;
```

//11. feladat Emelje az árakat 20 %-kal, majd írja ki az áruk nevét és új árát!

```
cout << "\\nAz áruk adatai 20%-os emelés után:" << endl;
cout.setf(ios::right);
for (i = 0; i < aruszam; i++)
{
    bolt[i].ar = bolt[i].ar * 1.2; // 20%-os emelés

    cout << setw(20) << bolt[i].arunev << "\\t" << bolt[i].ar << "Ft\\t"
    << bolt[i].db << "db" << endl;
}
return 0;
}
```

Feladat (rövidebben)

```
include <iostream>
#include<iomanip>
#include<string>
#include<ctime>
#include<fstream>
using namespace std;
struct Aru
{
    string arunev;
    int ar, db;
};
int main()
{
    Aru bolt[15];
```

```
int i, legdragabb, ertek = 0, atlagar = 0, aruszam;
srand((unsigned)time(NULL));
// adatbekérés - különböző áruk darabszámának bekérése
ellenőrzöten
do
{
    cout << "Adja meg hany fele aru van a boltban ! (1-15)" << endl;
    cin >> aruszam;
    cin.ignore();
    if (arszam < 1 || arszam>15) cout << "Hibas adat, az aruk
szama    1-15 lehet!" << endl;
} while (arszam < 1 || arszam>15);
```


Feladat (rövidebben)

```
// adatbekérés - különböző áruk neve, darabszáma és ára
for (i = 0; i < aruszam; i++)
{
    cout << "Adja meg az " << i + 1 << " . aru nevet\n";
    getline(cin, bolt[i].arunev);
    do
    {
        cout << "Adja meg az aru arat (100-5000Ft)\n";
        cin >> bolt[i].ar;
        cin.ignore();
        if (bolt[i].ar < 100 || bolt[i].ar > 5000)
            cout << "Hibas adat, az aruk ara 100-5000Ft lehet!" << endl;
    } while (bolt[i].ar < 100 || bolt[i].ar > 5000);
    bolt[i].db = rand() % 10 + 1;
}
```

```
system("cls");
int otneltobb = 0;
cout.setf(ios::left); // 5. feladat eredmények kiírása
    "táblázatosan"
for (i = 0; i < aruszam; i++)
{
    cout << setw(20) << bolt[i].arunev << bolt[i].ar << "Ft" << "\t"
    << bolt[i].db << "db" << endl;
    ertek = ertek + bolt[i].ar * bolt[i].db; //6. egész árukészlet értéke
    atlagar = +bolt[i].ar; // 7. feladat - átlagár kiszámítása
    if (bolt[i].db > 5) otneltobb++; //8. hány áruból van több mint öt
    if (bolt[i].ar > bolt[legdragabb].ar) legdragabb = i; //9. feladat-
    írja ki a legdrágább áru nevét és árát!
}
```

Feladat (rövidebben)

```
cout << "\nAz arukeszlet ossz ereteke: " << ertek << "Ft" << endl;
cout << "\nAz aruk atlagara: " << atlagar/aruszam << "Ft" << endl;
cout << endl << otneltobb << " arubol van otnel tobb a
keszleten." << endl;
cout << "\nA legdragabb aru adatai: \n" << setw(20)
<< bolt[legdragabb].arunev << bolt[legdragabb].ar << "Ft" << "\t"
<< bolt[legdragabb].arunev << "\t" << bolt[legdragabb].ar << endl;
```

//10. feladat Lista kiírása fájlba

```
ofstream keszlet("aktualis.txt");
if (keszlet.is_open())
{
    for (i = 0; i < aruszam; i++)
    {
        keszlet << bolt[i].arunev << "\t";
        keszlet << bolt[i].ar << "\t";
        keszlet << bolt[i].db << endl;
    }
    keszlet.close();
    cout << "\nA lista sikeresen elkeszult." << endl;
}
else cout << "\nA lista létrehozása sikertelen." << endl;
```

Feladat (rövidebben)

//11. feladat Emelje az árakat 20 %-kal, majd írja ki az áruk nevét és új árát!

```
cout << "\nAz aruk adatai 20%-os emeles utan:" << endl;
cout.setf(ios::right);
for (i = 0; i < aruszam; i++)
{
    bolt[i].ar = bolt[i].ar * 1.2; // 20%-os emelés
    cout << setw(20) << bolt[i].arunev << "\t" << bolt[i].ar << "Ft\t" << bolt[i].db << "db" << endl;
}
return 0;
}
```