

Karakterláncok (sztringek) I

Kormányos Andor

Komplex Rendszerek Fizikája Tanszék

2023 október 24.

A C++ -ban létezik egy adatstruktúra a karakterláncok kezelésére

- nem beépített adattípus, mint pl a `double` stb
- lényegében hasonló a `std::vector`-hoz, néhány extra funkcióval
- használatához szükséges egy `header` és jelezni kell, hogy melyik `namespace`-ben van definiálva

```
#include <string>
using std::string;
```

Karakterláncok (stringek) a C++ nyelvben

A C++ -ban létezik egy adatstruktúra a karakterláncok kezelésére

- nem beépített adattípus, mint pl a `double` stb
- lényegében hasonló a `std::vector`-hoz, néhány extra funkcióval
- használatához szükséges egy `header` és jelezni kell, hogy melyik `namespace`-ben van definiálva

```
#include <string>
using std::string;
```

Sztringek inicializálása

```
string s1; // ures string
string s2= "hello"; // sztring literallal
string s3("hello") // ugyanaz, mint fentebb
string s3(5, 'c'); // 5db c karakter egymás után: ccccc
```

Figyelem!

- a string literalokat dupla idézőjel jelzi, pl.: `"alma"`
- a karakter literalokat szimpla idézőjel jelzi, pl.: `'c'`

A beolvasás és kiírás ugyanúgy, mint a beépített típusoknál

```
int main()
{
    string szo;
    cin >> szo;
    cout << szo << endl;
    return 0;
}
```

Figyelem!

- a `>>` nem veszi figyelembe, ha a sztring elején vannak szóközök
- és a soremelést sem
- a következő szóköz karakterig olvas
- tehát ha a konzolablakba "Hello world!"-t írunk, akkor a kimenet "Hello"

Indexek használata

- `str[indx]` az `str` karakterlánc *indx*-ik elemét jelöli
- *indx* lehetséges értékei: $0 \leq \textit{indx} < \textit{str.size}()$
- *indx* típusa: `size_t`

További műveletek

- `s.empty()`: `true`, ha a sztring üres, egyébként `false`
- `s1+s2` : két sztring összefűzése
- `s1=s2`: `s2` átmásolódik `s1` -be
- `s1==s2`, `s1!=s2` stb