
Simbolių eilutės `string`

Esminiai dalykai

- Darbui su simbolių eilutėmis yra sukurta klasė (tipas) `string`. Failas, kuriame yra aprašyta klasė `string`, į programą įterpiamas sakiniu

```
#include <string>
```

- Simbolių eilučių tipo kintamieji (objektai) aprašomi taip pat, kaip ir kitų tipų kintamieji:

```
string eil, e1;
```

- Sukurtos eilutės yra tuščios. Jas aprašant galima priskirti reikšmes:

```
string eil = "Jau moku programuoti C++ kalba";
```

Darbui su eilutėmis naudojami operatoriai (1)

- Priskyrimo (=):

```
string s1, s2;  
string s = "12";  
char c = 'z';  
s1 = s; s2 = c;
```

- Rezultatas – s1 reikšmė ("12") ir s2 reikšmė ("z").

Darbui su eilutėmis naudojami operatoriai

(2)

- Sudėties (+). Sudedant eilutes s1 ir s2, prie pirmos eilutės pabaigos pridedama antros eilutės kopija. Galima sudėti dvi C++ eilutes arba C++ eilutę ir simbolį.

```
string s1 = "a", s2 = "b";  
string s = "12"; char c = 'z';  
s1 = s1 + s; s2 = s2 + c;
```

- Rezultatas – s1 reikšmė ("a12") ir s2 reikšmė ("bz").

Darbui su eilutėmis naudojami operatoriai

(3)

- Lyginimo (==, !=, < ir kt.).
 - Lyginami eilutes sudarančių simbolių kodai.
 - Dvi eilutės lygios, jei jos yra vienodo ilgio ir sutampa visų jų simbolių kodai poromis.
 - C++ eilutę galima lyginti tik su kita C++ eilute.
-

Įvesties ypatumai

- Operatorius >> įveda į string tipo eilutę simbolių seką iki tarpo arba iki eilutės pabaigos simbolio '\n'.
 - Norint įvesti dalį simbolių eilutės (pvz., ne iki tarpo), reikia naudotis globaliaja funkcija `getline()`.
-

Pavyzdys

Tarkime, faile `fd` yra eilutė

```
Dangaus laiptelis * Andersen
```

Pavyzdys. Veiksmai ir jų rezultatai, rodomi ekrane

Veiksmai	Rezultatai ekrane
<pre>string eil fd >> eil; // fd – įvesties srautas cout << eil;</pre>	Dangaus
<pre>string eil getline(fd, eil); // fd – įvesties srautas cout << eil;</pre>	Dangaus laiptelis * Andersen
<pre>string eil getline(fd, eil, '*'); // fd – įvesties srautas cout << eil;</pre>	Dangaus laiptelis

Klasės `string` funkcijos, dažnai naudojamos darbui su eilutėmis

Funkcijos prototipas	Paiškinimai
<code>size_type max_size() const;</code>	Grąžina elementų, kurie gali būti eilutėje, skaičių. <i>Pavyzdys</i> <pre>string s("C++ kalba"); int n = s.max_size();</pre> Rezultatas – n reikšmė: 4294967281.
<code>size_type length() const;</code>	Grąžina eilutės ilgį (simbolių skaičių eilutėje). <i>Pavyzdys</i> <pre>string s("C++ kalba"); int n = s.length();</pre> Rezultatas – n reikšmė: 9.
<code>const char * c_str();</code>	Grąžina rodyklę-konstantą į C eilutę, kurios reikšmė sutampa su eilute, kuriai taikoma ši funkcija. Naudojama failų atidarymo sakiniuose. <i>Pavyzdys</i> <pre>string s = "Duomenys.txt"; ifstream fd(s.c_str());</pre> Rezultatas – eilutė s, konvertuota į C eilutę.
<code>string substr(size_type k, size_type n = npos);</code>	Iš eilutės, kuriai taikoma ši funkcija, kopijuojama n simbolių, pradedant k-uuju. <i>Pavyzdys</i> <pre>string s = "Programavimo kalba C++"; string s1 = "Matematika"; s1 = s.substr(13, 5);</pre> Rezultatas – nauja eilutės s1 reikšmė: "kalba".