

Satrlar bilan ishlash (1-qism)

Reja:

- C++ da satrlar haqida
- Satr uzunligini aniqlash
- Satr uzunligini o'zgartirish
- Satrli o'zgaruvchilarga qiymat berish
- Satrning davomiga satr ulash
- Satrning biror qismini o'chirish
- Amaliy mashqlar

C++ da satrlar

C++ tilida satrlarning ifodalanishi ikki xil:

- ✓ `char array`
- ✓ `string class obyekti`

Satr char turidagi belgilar massivi sifatida qaralganda bu belgilar ketma-ketligi satr terminatori deb nomlanuvchi nol kodli belgi bilan tugaydi ('\0').

Odatda, nol-terminator bilan tugaydigan satrlarni ASCIIZ – satrlar deyiladi.

Satr o'zgaruvchilari e'lon qilinishida boshlang'ich qiymatlar qabul qilishi mumkin. Bu holda kompilyator avtomatik ravishda satr uzunligi hisoblaydi va satr oxiriga nol-terminatorni qo'shib qo'yadi:

```
char dayOfWeek[]="Juma";
```

Ushbu e'lon quyidagi e'lon bilan ekvivalent:

```
char dayOfWeek[]= {'J','u','m','a','\0'};
```

string

C++ da satrlar bilan ishlashni qulaylashtirish uchun **string** sinfi kiritilgan.

string sinfi satrlarida satr oxirini **'\0'** belgisi belgilamaydi.

Standart kutubxonadagi **string** class idan foydalanish uchun `<string.h>` sarlavha faylini dasturga qo'shish kerak.

```
string str = "Bu yozuv satr hisoblanadi.";
cout<<str<<endl;
```

```
Bu yozuv satr hisoblanadi.
```

Satrlar bilan ishlovchi asosiy funksiyalar bilan tanishib chiqamiz.



Satr uzunligini aniqlash

Satrning uzunligini aniqlash uchun `length()` funksiyasidan foydalaniladi.

```
unsigned int size() const;  
unsigned int length() const;  
unsigned int max_size() const;  
unsigned int capacity() const;  
bool empty() const;
```

```
// SATR O'LCHAMI  
// SATR ELEMENTLARI SONI  
// SATRNING MAKSIMAL UZUNLIGI  
// SATR EGALLAGAN XOTIRA HAJMI  
// SATRNING BO'SHLIGINI ANIQLASH
```

*main.cpp x

```
1  #include <iostream>
2  #include <string.h>
3
4  using namespace std;
5
6  int main()
7  {
8      string s;
9
10     cout << "Satr kiriting" << endl;
11     getline(cin, s);
12
13     cout << "Siz kiritgan satr " << s.length() << " ta belgidan iborat" << endl;
14     cout << "Siz kiritgan satr " << s.size() << " ta belgidan iborat";
15
16     return 0;
17 }
18
```

```
Satr kiriting  
Assalomu alaykum  
Siz kiritgan satr 16 ta belgidan iborat  
Siz kiritgan satr 16 ta belgidan iborat
```


Satr uzunligini o'zgartirish

Satr uzunligini o'zgartirish uchun `resize()` funksiyasidan foydalaniladi.

```
void string ::resize (size_type num)
```

```
void string ::resize (size_type num, char c )
```



main.cpp x

```
1  #include <iostream>
2  #include <string.h>
3
4  using namespace std;
5
6  int main()
7  {
8      size_t n;
9
10     string str ("Men kelajakda backend dasturchi bo'lishni xohlayman");
11     cout << str << endl;
12
13     // SATR UZUNLIGINI ANIQLASH
14     n = str.size();
15
16     // SATR UZUNLIGINI 2TA BELGIGA UZAYTIRISH
17     str.resize (n+2, '+');
18     cout << str << endl;
19
20     // SATR UZUNLIGINI O'ZGARTIRISH
21     str.resize(14);
22     cout << str << endl;
23
24
25     return 0;
26 }
```

```
Men kelajakda backend dasturchi bo'lishni xohlayman  
Men kelajakda backend dasturchi bo'lishni xohlayman++  
Men kelajakda
```

```
Process returned 0 (0x0)   execution time : 0.016 s  
Press any key to continue.
```

Satr bo'shligini tekshirish

`bool empty()` - funksiyasi satrni bo'shligini tekshirish uchun ishlatiladi. Agar satr bo'sh bo'lsa, `true` qiymat qaytaradi.

Satrli o'zgaruvchilarga qiymat berish

```
// SATRGA str O'ZGARUVCHISIDAGI SATRNING TO'LIQ NUSXASINI OLISH|  
string& assign ( const string &str);
```

```
// SATRGA str O'ZGARUVCHISIDAGI pos O'RNIDAGI BELGISIDAN BOSHLAB n ta BELGI NUSXASINI OLISH  
string& assign ( const string &str, size_t pos, size_t n);
```

```
// STRING TURIDAGI SATRGA CHAR TURIDAGI SATRNING NTA BELGISINI NUSXALASH|  
string& assign ( const char* s, size_t n);
```


Satrning davomiga satr ulash

Satrning davomida boshqa satr ulash uchun `append()` funksiyasidan foydalaniladi.

```
string& append (const string& str);  
string& append (const string& str, size_t pos, size_t n);  
string& append (const char* s, size_t n);
```

`char array` toifasidagi satrni `string` toifasiga o'tkazish

main.cpp x

```
1  #include <iostream>
2  #include <string.h>
3
4  using namespace std;
5
6  int main()
7  {
8
9      char c[100];
10     string s = "Salom Dunyo";
11
12     s.clear();           // s ning qiymati tozalandi
13
14     cout << "Satrni kiriting: |" << endl;
15     cin.getline(c, sizeof(c));
16
17     // s satriga char toifasidagi c satrini nusxasini olish
18     s.assign(c, strlen(c));
19
20     cout << s << endl;
21
22     return 0;
23 }
24
```

`string` toifasidagi satrni `char array` toifasiga o'tkazish

`string` toifasidagi satrni `char array` toifasiga o'tkazish uchun `c_str()` yoki `data()` funksiyalaridan foydalanish mumkin.

```
const char* c_str() const;  
const char* data() const;
```

Satrning biror qismini o'chirish

```
erase (unsigned int pos = 0, unsigned int n = pos);|
```

`erase()` funksiyasi satrni `pos` o'zgaruvchisida ko'rsatilgan o'rnidan boshlab `n` ta belgini o'chiradi.

Agar nechta belgi o'chirilishi **n** ko'rsatilmagan bo'lsa,
pos o'zgaruvchisida ko'rsatilgan o'rnidan boshlab satr
oxirigacha o'chiriladi.

Agar **pos** va **n** ko'rsatilmagan bo'lsa, satr to'liq
o'chiriladi.

`void clear()` - funksiyasi satrni tozalash (to'liq o'chirish) uchun ishlatiladi.

Ama1iy mashqlar

S satr berilgan. S satrida uchragan har bir belgini 2 marta orttiruvchi dastur tuzing.

C belgisi va S satri berilgan. S satrida uchragan har bir C belgisini 2 marta orttiruvchi dastur tuzing.

2 ta satr berilgan. Ularni qisqa+uzun+qisqa shaklida birlashtirilishidan hosil bo'lgan satrni hosil qiling.

Masalan:

`s1="salom", s2="olam" => natija: s3="olamsalomolam".`

N1, N2 natural sonlari va s1, s2 satrlari berilgan. s1 satrning dastlabki N1 ta belgisidan va s2 satrning oxirgi N2 ta belgisidan iborat yangi satr hosil qiling.

S1 va S2 satrlari berilgan. Agar S2 satri S1 satrida bo'lsa true, aks holda false chiqaruvchi dastur tuzing.

S1 va S2 satrlari berilgan. S2 satrini S1 satrida takrorlanishlar sonini chiqaruvchi dastur tuzing.

S1 va S2 satrlari berilgan. S1 satrida birinchi uchragan S2 satrini o'chirib tashlovchi dastur tuzing. Agar S1 satrida S2 satri uchramasa S1 o'zgarishsiz qolsin.

S1, S2 va S3 satrlari berilgan. S1 satrida birinchi uchragan S2 satrini S3 satriga o'zgartiruvchi dastur tuzing.

**E`tiboringiz uchun
rahmat!**