**C++ STRINGI**

**Ni pravi kontejner, ni pravi template, NI običajen razred**

**string je TEMPLATE Z DEFINIRANIM PARAMETROM:**

**typedef basic\_string <char> string;**

Deklaracija praznega niza:

string ime\_niza;

Deklaracija in inicializacija:

string ime\_niza("znaki\_v\_nizu");

string ime\_niza="znaki\_v\_nizu";

še možne deklaracije in inicializacije:

string niz2(niz1); // kopija stringa niz1

string niz3(3, '?');// niz3 dobi 3 znake ? in ima vrednost: "???"

Primeri:

#include <iostream>

#include <string>

using namespace std;

int main()

{

string a; //prazen string

getline(cin,a); //cin bere samo do presledka

string b("kr eno besedilo");

string c="tretji string";

cout<<a<<endl;

cout<<b<<endl;

cout<<c<<endl;

cout << "Hello World!" << endl;

return 0;

}

Za string so nekateri operatorji operatorji že preobloženi:

operator = //namesto strcpy()

niz1=niz2;

**operator +=**

niz += "dodaj";

**operator +** //namesto strcat;

niz = niz1 + niz2;

niz1=niz1+ "krneki";

lahko uporabimo metodo **append()**:

niz1=niz1.append("krneki");

Primer:

#include <iostream>

#include <string>

using namespace std;

int main()

{

string a;

string b("kr eno besedilo");

string c="tretji string";

a=c+" ";

a=a.append(b);

cout<<a<<endl;

cout<<b<<endl;

cout<<c<<endl;

cout << "Hello World!" << endl;

return 0;

}

Preobloženi so tudi primerjalni operatorji (==, >, <, <=):

if (niz1 == niz2)… //namesto strcmp

lahko pa uporabimo tudi metodo **compare():**

niz1.compare(niz2) //vrne enako kot strcmp

#include <iostream>

#include <cstdio>

#include <string>

using namespace std;

int main()

{string a,b;

getline(cin,a);

getline(cin,b);

if (a.compare(b)==0)

cout<< "niza sta enaka"<<endl;

else if (a.compare(2,3,b)>0)

cout<<"v prvem nizu so znaki od indexa 2 do 5 po abecedi večji od niza b"<<endl;

else if(a.compare(2,3,b,4,3)==0)

cout<<"v prvem nizu so znaki od indexa 2 do 5 po abecedi enaki znakom na indeksih od 4 do 7 niza b"<<endl;

return 0;

}

kopiranje iz stringa v char \*:

//namesto strcpy uporabljamo metodo **copy():**

primer:

#include <iostream>

#include <cstdio>

#include <string>

using namespace std;

int main(){

int dolg;

string a = "en string";

char b[20];

dolg=a.copy(b,2,0);//vrne število kopiranih znakov; 2 znaka od indexa 0

b[dolg]='\0'; //dodamo znak za konec niza

cout<<b;

return 0;

}

za dolžino stringa se uporabljata **metodi** length() in size():

cout<<niz1.length();

cout << niz1.size();

Dostopanje do posameznega znaka:

niz[i]

niz.at(i) //v primeru, da se nahaja izven samega string, vrže izjemo

metode:

**resize():**

#include <iostream>

#include <string>

using namespace std;

int main ()

{

string niz ("Krneki");

cout << niz << endl;

niz.resize (15,'.'); //dodamo toliko znakov ., koliko manjka do dolžine 15

cout << niz << endl;

niz.resize (3);

cout << niz << endl;;

return 0;

}

**reserve():**

#include <iostream>

#include <string>

using namespace std;

int main ()

{

string niz ("Krneki");

cout << niz << endl;

niz.reserve(15);

cout << niz << endl;;

return 0;

}

**capacity**():

#include <iostream>

#include <string>

using namespace std;

int main ()

{

string niz ("Krneki");

cout << niz << endl;

cout<<niz.capacity();

niz.resize(15);

cout << niz << endl;

cout<<niz.capacity();

return 0;

}

**insert() //**metoda, ki vstavi oz. vrine v niz:

#include <iostream>

#include <string>

using namespace std;

int main ()

{

string niz ("Krneki");

cout << niz << endl;

niz.insert(3,2,'\*'); //na index 3 vstavi 2 znaka '\*'

cout << niz << endl;

niz.insert(2,"vstavljeno"); //na index 2 vstavi besedilo

cout << niz << endl;

niz.insert(4,"se nekaj",2,5); //na index 4 vstavi 5 znakov, ki se začnejo na indeksu 2 niza "se nekaj"

cout << niz << endl;

return 0;

}

**erase():**

#include <iostream>

#include <string>

using namespace std;

int main ()

{

string niz ("Krneki");

cout << niz << endl;

niz.erase(2,3); //zbriše 3 znake od indexa 2 naprej

cout << niz << endl;

return 0;

}

**clear(), empty():**

#include <iostream>

#include <string>

using namespace std;

int main ()

{

string niz ("Krneki");

cout << niz << endl;

if (niz.empty()) //preverjanje, ali je niz prazen

cout<<"niz je prazen"<<endl;

else

cout<<"niz ni prazen"<<endl;

niz.clear(); //brisanje vseh znakov iz niza

cout << niz<<endl<<niz.size() << endl;

if (niz.empty())

cout<<"niz je prazen"<<endl;

else

cout<<"niz ni prazen"<<endl;

return 0;

}

**substr():**

#include <iostream>

#include <string>

using namespace std;

int main ()

{

string niz ("Krneki");

cout << niz << endl;

string novi=niz.substr(2,3); //kopira podniz dolžine treh znakov od indexa 2

cout << novi << endl;

return 0;

}

**find() //išče znak od podniz:**

#include <iostream>

#include <string>

using namespace std;

int main ()

{

string niz ("Krneki");

cout << niz << endl;

cout<<"znak n se nahaja na indexu "<<niz.find('n'); //vrne index

return 0;

}

#include <iostream>

#include <string>

using namespace std;

int main ()

{

string niz ("Krneki");

cout << niz << endl;

cout<<"podniz ek se zacne na indexu "<<niz.find("ek");

return 0;

}