Vaje 27

Klavdija Lunar

1.6.1998

Lastnosti razreda `Cniz` so:

- generator,
- destruktor,
- vrednost,
- dolžina.

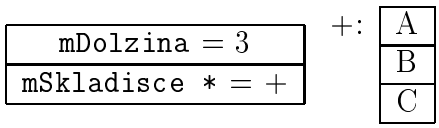
Ogledali smo si tudi testni program, ki si ga lahko ogledaš na:

`Z:\PREDAVANJA\POLDE\test\Cniz\TestCniz.cpp`

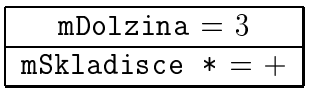
KAJ POMENI PRENOS PO REFERENCI? To pomeni, da pokličeš funkcijo in ji podaš *original*.

KAJ POMENI PRENOS PO VREDNOSTI? To pomeni, da pokličeš funkcijo in ji podaš *kopijo* originala. Pri prenosu po vrednosti se požene *prepisovalni kreator* (*copy-creator*), ki original ,,prekopira" v kopijo.

KAKO KOPIRAMO? Original:



skopiramo oz. naredimo prenos po vrednosti v:



`mDolzina` je število znakov v nizu, `mSkladisce` pa je kazalec na polje znakov. Oba se prepišeta takšna, kot sta. Ko sedaj poženemo ,,destruktor" na kopiji, le-ta uniči tudi niz `ABC`, pa čeprav je še vedno del originala – in ga zato ne bi smel uničiti.

Rešitev je, da kopiranje ne prepiše samo `mDolzina` in `mSkladisce`, ampak tudi niz `ABC`. To dosežemo z uporabo zgoraj omenjenega prepisovalnega kreatorja. V definicijski modul dodamo:

```
  CNiz(CNiz & original);
/* DESC:   Prepise original v niz this v celoti -- vkljucno z nizom
           (vsebino skladisca), na katerega kaze mSkladisce.
   PARMS:  nekaj -- prepisani element (po referenci).
   RESULT: /
   PRE:
   POST:   (mDolzina = nekaj.mDolzina) AND
           (*mSkladisce = *staro.mSkladisce)
*/
```

Podrobnosti so v:

`Z:\PREDAVANJA\POLDE\code\Cniz\Cniz.h`

Uporaba `operator`-jev

Želeli bi dobiti i-ti element niza. Z dosedanjim znanjem bi znali definirati metodo `Element` s parametrom indeks, ki bi vrnila zahtevani element niza. Bolj razumljivo bi bilo, če bi lahko zapisali:

```
Cniz mojNiz;

printf("3. element mojega niza je %c", mojNiz[3]);
```

To dosežemo z uporabo *operatorjev*. Sintaksa je naslednja:

$$charoperator[](intindeks)$$

Podrobnosti so na:

`Z:\PREDAVANJA\POLDE\code\Cniz\Cniz.h`

`Z:\PREDAVANJA\POLDE\code\Cniz\Cniz.cpp`

`Z:\PREDAVANJA\POLDE\test\Cniz\TestCniz.cpp`