
Operatori *new* i *delete*

- ❖ Alokatorska funkcija *new* može se preklopiti za neku klasu *T* kao statička funkcija članica te klase (čak i ako se ne deklarise kao *static*, uvek je implicitno statička):

```
void* T::operator new (size_t size, placement_params)
```

```
void* T::operator new [] (size_t size, placement_params)
```

- ❖ Ove funkcije mogu da se preklope u svim oblicima u kojima postoje i takve ugrađene funkcije
- ❖ Ove funkcije ne treba da pozivaju konstruktore eksplicitno; konstruktor se uvek poziva kao drugi korak u izvršavanju izraza *new* čiji samo jedan korak poziv ove alokatorske funkcije
- ❖ Ove funkcije imaju zadatak da pronađu “sirov” memorijski prostor zadate veličine i ništa više, nikako da od njega prave objekat (što radi konstruktor)
- ❖ Upravo zato i vraćaju pokazivač tipa *void** koji sadrži adresu alociranog prostora i upravo zato su one statičke funkcije članice klase, jer se pozivaju pre pravljenja objekta te klase
- ❖ Ako klasa ima ovakvu funkciju, ona će biti pozivana u izvršavanju prvog koraka izraza *new* kada se alociraju objekti, odnosno nizovi objekata te klase
- ❖ Ovakva funkcija može alocirati dinamičke objekte nekog drugog tipa ili na drugi način rukovati prostorom

Operatori *new* i *delete*

- ❖ Analogno, dealokatorska funkcija *delete* može se preklopiti za neku klasu *T* kao statička funkcija članica te klase (čak i ako se ne deklariše kao *static*, uvek je implicitno statička):

void T::operator delete (void*)

void T::operator delete [] (void*)

- ❖ Ove funkcije mogu da se preklope u još nekim dostupnim oblicima (npr. onim kojim primaju veličinu prostora koji se dealocira)
- ❖ Ove funkcije ne treba da pozivaju destruktore eksplicitno; destruktor se uvek poziva kao korak u izvršavanju izraza *delete*
- ❖ Ove funkcije imaju zadatak da memorijski prostor na zadatoj adresi proglase slobodnim i ništa više
- ❖ Upravo zato i primaju pokazivač tipa *void** koji sadrži adresu prostora koji treba dealocirati (a ne pokazivač na objekat klase, jer je objekat već uništen)
- ❖ Ako klasa ima ovakvu funkciju, ona će biti pozivana u izvršavanju drugog koraka izraza *delete* kada se dealociraju objekti, odnosno nizovi objekata te klase