
Operatori *new* i *delete*

- ❖ Treba primetiti da ovaj problem, za razliku od problema curenja memorije, nije posledica bilo kakve greške u programiranju, već se dešava tokom regularnog izvršavanja sasvim regularnih programa
- ❖ Ovaj problem je posledica toga što se objekti različite veličine alociraju u istom zajedničkom prostoru, pa se u oslobođeni prostor jedne veličine smešta objekat manje veličine, i tom prilikom nastaje fragment slobodne memorije
- ❖ Rešenje ovog problema tzv. *kompakcijom* (pomeranjem alociranih objekata tako da se oni spoje, a slobodan prostor fuzioniše na kraju) nije primenjiv, jer objekti mogu da imaju pokazivače u sebi i da na njih ukazuju pokazivači za koje se ne zna gde su u memoriji; osim toga, takvo rešenje trošilo bi značajno režijsko vreme zbog kopiranja mnogo delova memorije
- ❖ Ovo može da bude motiv za preuzimanje kontrole nad alokacijom objekata neke klase, recimo tako što se za njih odvoji poseban niz pregradaka (slotova) iste veličine: kada se u oslobođeni prostor jedne veličine smešta objekat iste veličine, nema fragmenta

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 deklariše 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