

## DINAMI^KI OBJEKTI

Dinami~ka alokacija i objekti stoje u dvojnomo odnosu: dinami~ke promenljive mogu biti atributi objekata, ali i sami objekti mogu biti dinami~ki.

Dinami~ki atributi objekata (ako nisu i sami objekti) zah~evaju isti tretman kao i ostale dinami~ke promenljive i to bez obzira na to da li je sam objekat dinami~ki ili nije: moraju biti kreirani operacijom *New*, odnosno poni{teni operacijom *dispose*. Ako klasa ima dinami~kih atributa logi~no je da se predvidi metoda za inicijalizaciju (jedna ili vi{e) u kojoj se otvara memorijski prostor za njih i (eventualno) upisuju po~etne vrednosti. Ova procedura obi~no se zove *Init* po ugledu na konstruktor. Ukoliko klasa ima inicijalizaciju ona se u celosti sme{ta u ovu proceduru.

Prema tome, nezavisno od toga da li klasa ima virtuelnih metoda ili ne procedura *Init* za inicijalizaciju mo`e da sadr`i:

- Zadavanje po~etnih vrednosti stati~kim atributima
- Kreiranje dinami~kih atributa provedurom *New*
- Zadavanje po~etnih vrednosti kreiranim dinami~kim atributima.

Naravno, ako klasa ima virtuelnih metoda, procedura za inicijalizaciju realizuje se kao konstruktor.

Postojanje dinami~kih atributa u objektu, normalno, podrazumeva potrebu za operacijom dealokacije procedurom *dispose*. Kao i za alokaciju, za dealokaciju se u na~elu upotrebljava jedna procedura (iako ih mo`e biti i vi{e). U nju se sme{taju pozivi procedure *dispose* za sve dinami~ke attribute. Programeri joj obi~no daju ime *Done*.

Drugi - mnogo interesantniji - slu~aj je kada je sam objekat dinami~ki, tj. kada se u toku izvr{enja programa kreira u dinami~koj memoriji. Dinami~ki objekti se, kao i sve ostale dinami~ke promenljive, realizuju posredstvom pokaziva~a. Pokazi-

va~i na objekte se sintaksno ne razlikuju od drugih pokaziva~a. Naime, ako je deklarirana klasa Obj naredbom

```
type Obj = object
    . . .
end;
```

tada se pokaziva~ na instancu te klase sa imenom npr. pOB defini~e sa

```
var pO: ↑Obj;
```

Mnogo ~e{}e, me|utim, uz osnovnu klasu defini~e se i *tip pokaziva~a* na klasu i taj tip dodeljuje pokaziva~koj promenljivoj. Dakle, umesto gornje sekvence u praksi se obi~no pi~e

```
type PObj = ↑Obj;
var pO: PObj;
```

Tako|e je uobi~ajeno me|u programerima da naziv tipa pokaziva~a na objekat bude ime osnovne klase sa prefiksom P (Obj i PObj u primeru).

Pokaziva~ka promenljiva pO predstavlja dinami~ki objekat koji se kreira procedurom *New*, a poni~tava procedurom *dispose*. Ipak, situacija ovde nije tako jednostavna kao kod standardnih pokaziva~kih promenljivih: noty slo`enosti unose potencijalno postojanje virtuelnih metoda, zatim mogu}nost da dinami~ki objekat ima tako|e dinami~ke attribute i (naro~ito neprijatna) mogu}nost upotrebe polimorfizama.

Pre svega, dinami~ki objekat kreira se procedurom *New*. Ako on ima virtuelnih metoda koje, sa svoje strane, imperativno nala`u da se pri kreiranju odmah aktivira i konstruktor, tada procedura *New* u obi~noj formi ne bi mogla da obavi tra`eni posao. Usled toga, procedura *New* morala je biti modifikovana za