

Corso di laurea in Informatica Applicata

Fondamenti di Programmazione

Appello del 20/1/2006

ESERCIZIO 1

Dato il seguente programma:

```

prog {class A{
    public int x; public int y;
    public int Somma(B j) {
        (3)
        if (j.val) {int s= this.x+this.y;
        (4)
            return s;
        }
        else return 0;
        (5)
    }
}
class B{public boolean val;}}
(1)
{
boolean y=true;
A obg= new A();
B obbg= new B();
obg.x=3; obg.y=5;
obbg.val=y
(2)
in z=obj.Somma(obbg);
(6)
}
    
```

rappresentare graficamente:

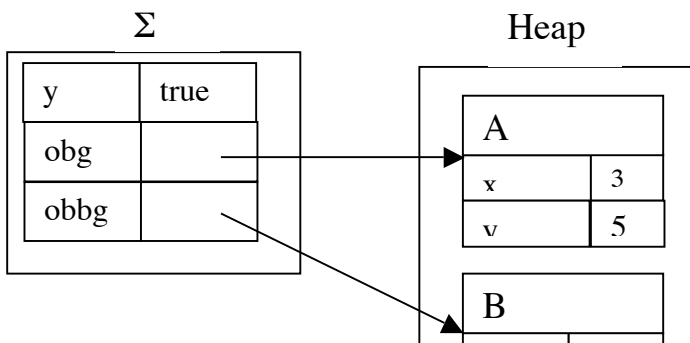
- I. l'ambiente delle classi al punto (1);
- II. lo stack di frames e lo heap dopo l'esecuzione dei comandi (2), (3),(4),(5) e(6).

Soluzione

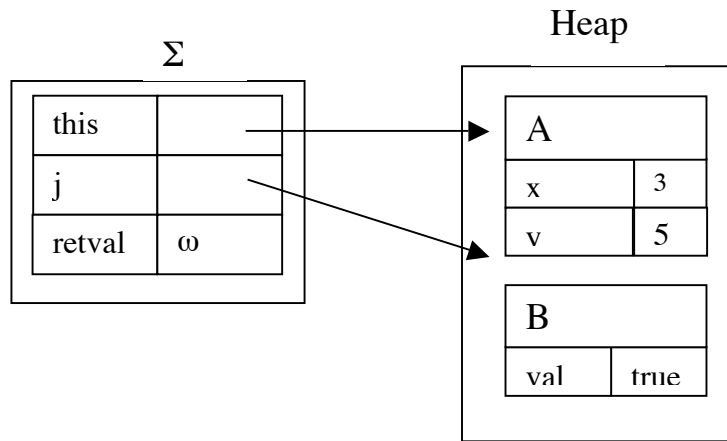
I.(1)

A	{(x,ω),(y, ω)}	Somma j if {int s=this.x+this.y; return s;} else return 0;
	{(val, ω)}	ω

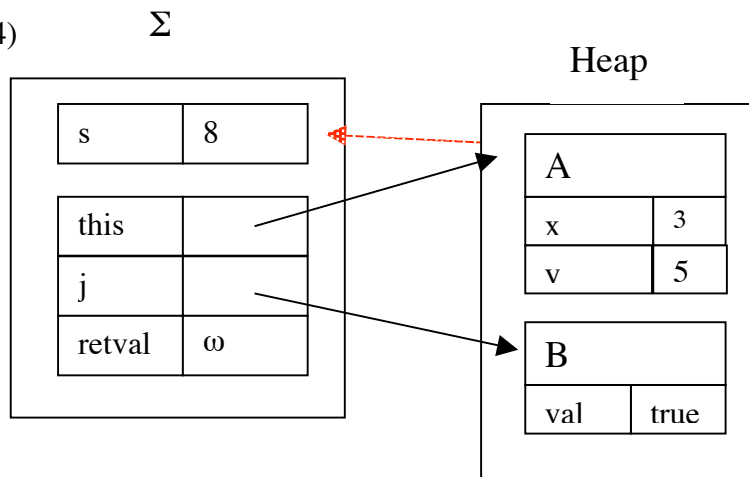
II(2)



(3)



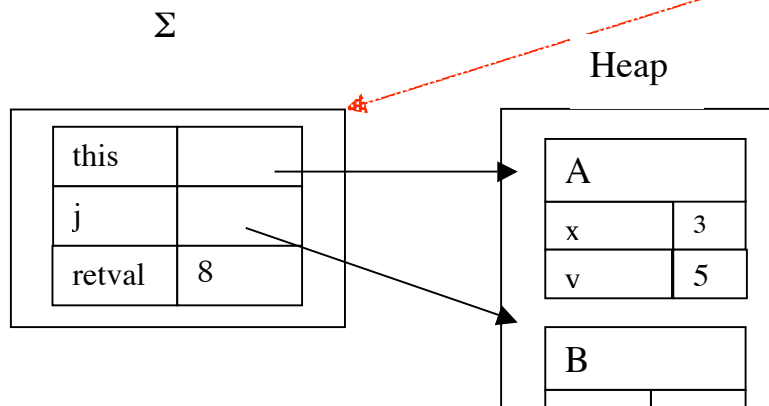
(4)

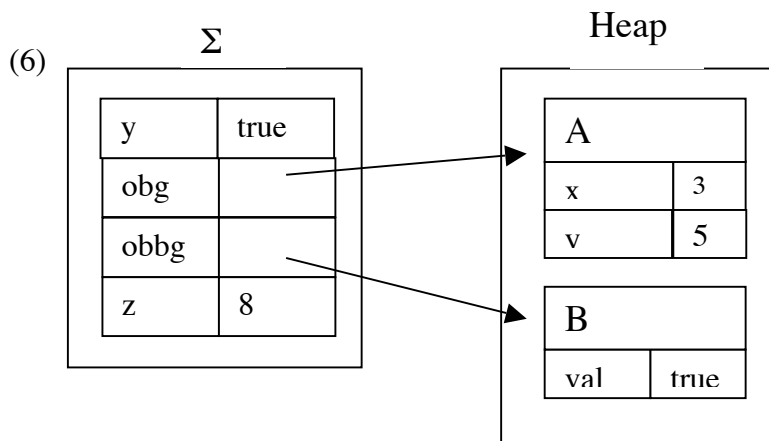


nuovo frame inserito
in testa allo stack a
causa del blocco del
ramo then dell'if

tale frame viene
eliminato quando si
esce dal blocco (5)

(5)





ESERCIZIO 2

Si supponga di estendere il linguaggio didattico consentendo di inizializzare le variabili d'istanza delle classi. Sintatticamente la dichiarazione di una variabile d'istanza avrà ora due casi:

$InstVarDecl ::= \mathbf{public} \text{ Type } Ide; \quad (1)$

$\mid \mathbf{public} \text{ Type } Ide = Exp; \quad (2)$

Il caso (1) previsto anche prima che non inizializza la variabile e il caso (2), che prevede l'inizializzazione. Tale caso può essere applicato solo a variabili d'istanza di tipo primitivo int, boolean o char specificando un valore costante (ad es. per int 4,88, -45 ecc, oppure per char 'a', o 'b' ecc. oppure boolean true o false). Si definisca la nuova regola semantica per $InstVarDecl$ nel caso (2) e si commenti per iscritto la soluzione scelta per la semantica dell'espressione Exp che calcola il valore iniziale.

Soluzione

$$\langle Exp, \langle \omega, \omega.\Omega, \omega \rangle \rangle \rightarrow_{com} \langle v, \zeta \rangle$$

$$\langle \mathbf{public} \text{ T } x = Exp; , \varphi \rangle \rightarrow_{istdecl} \varphi [v/x]$$

la semantica di Exp viene valutata in uno stato costruito appositamente dove $\rho_c = \omega$, $\sigma = \omega.\Omega$, $\zeta = \omega$ simile allo stato in cui viene valutato il corpo del programma ma in questo caso anche l'ambiente delle classi (ρ_c) è l'ambiente vuoto. Questa soluzione funziona ed è ragionevole nell'ipotesi in cui Exp è un valore costante (funziona anche nell'ipotesi che Exp sia un'espressione costante cioè $3+5*4$). Lo heap risultante è indicato con ζ per generalità ma nell'ipotesi di espressioni o valori costanti, lo heap ζ risulterà necessariamente vuoto cioè $\zeta = \omega$. Nulla viene perso quindi ignorandolo.

