

Tecnologia ad oggetti

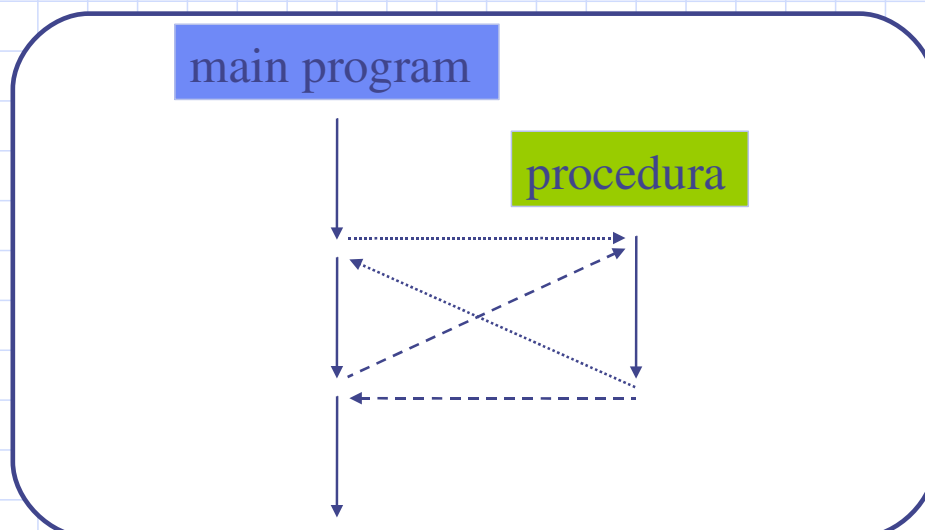
Oggetti:

- ❑ Origine del concetto e Incapsulamento
- ❑ Definizione
- ❑ Operazioni e attributi
- ❑ Rappresentazione grafica

Incapsulamento: *definizione generale*

Incapsulamento: raggruppamento di idee correlate in un'unica unità cui è possibile fare riferimento con un singolo nome.

Es. sottoprogramma (funzione, procedura, ...)



Incapsulamento *orientato agli oggetti*

Incapsulamento orientato agli oggetti: quando le **operazioni** (o **metodi**) e gli **attributi** (o **proprietà**) che rappresentano lo stato vengono raccolti in un tipo di oggetto in modo che lo stato sia accessibile o modificabile solo attraverso l'interfaccia fornita dall'incapsulamento.

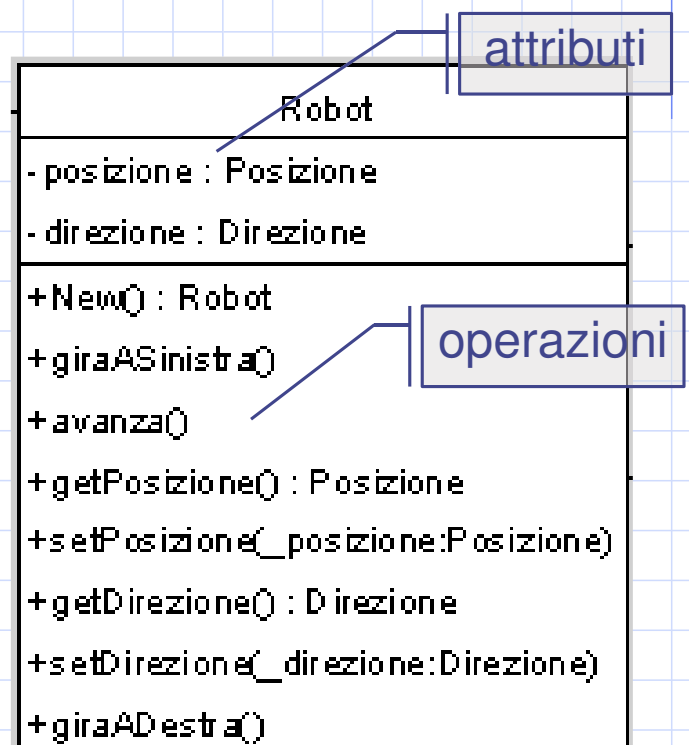
Es. oggetto di tipo Robot ...

Programmazione ad Oggetti - © S. Cicerone, G. Di Stefano

Oggetto: *definizione*

Un oggetto è costituito da un insieme di **attributi** e da un insieme di **operazioni**

- **direzione:** la direzione corrente del robot
- **giraASinistra ():** ruota a sinistra l'oggetto robot di 90 gradi a sinistra
- **avanza ():** sposta in avanti di una posizione il robot



Programmazione ad Oggetti - © S. Cicerone, G. Di Stefano

Oggetto: operazioni ed attributi

Operazioni:

- procedure o funzioni che normalmente risultano visibili e richiamabili da altri oggetti.
- permettono di modellare il **comportamento** dell'oggetto.

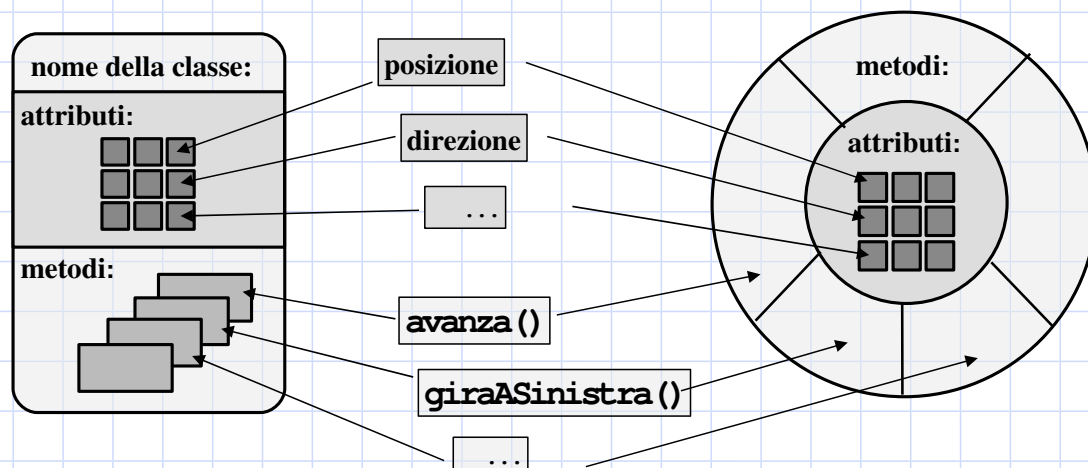
Attributi:

- rappresentano le informazioni che un oggetto "si ricorda", ovvero lo **stato** dell'oggetto.
- Solo le operazioni di un oggetto possono accedere ad essi ed aggiornarli.
- Nessun altro oggetto può accedere a un attributo accedendo direttamente alle variabili sottostanti che lo implementano; esso dovrà, infatti, ricorrere ad una delle operazioni dell'oggetto.

Programmazione ad Oggetti - © S. Cicerone, G. Di Stefano

Oggetti e Incapsulamento: rappresentazione grafica

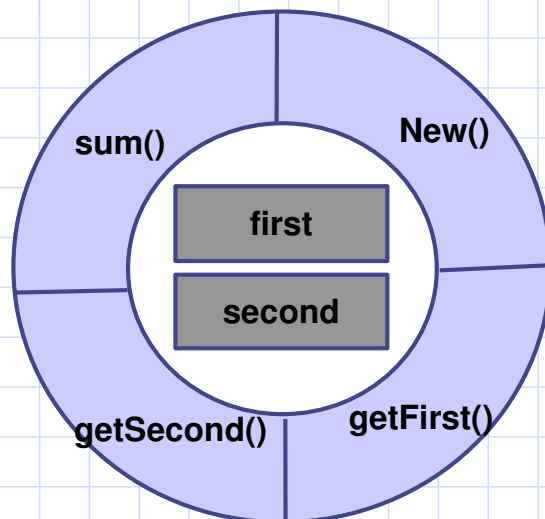
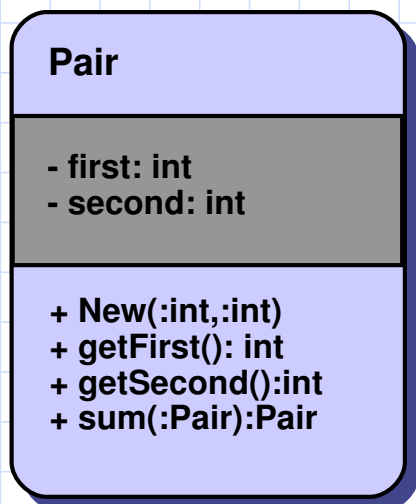
Dal momento che **solo** le operazioni dell'oggetto possono leggere e aggiornare gli attributi di quest'ultimo, queste operazioni formano un anello di protezione attorno al nucleo centrale delle variabili implementate all'interno dell'oggetto



Programmazione ad Oggetti - © S. Cicerone, G. Di Stefano

Oggetti e Incapsulamento: *un esempio*

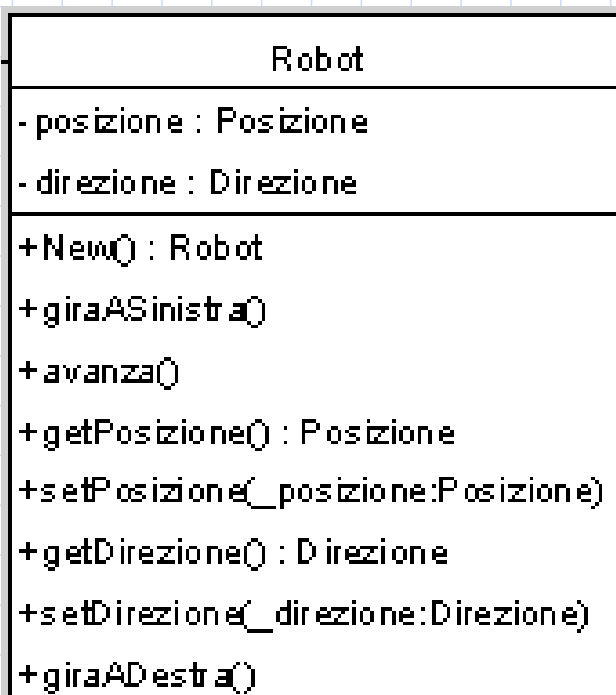
La classe Pair (coppia) prevede due interi come attributi accessibili attraverso i metodi `getFirst()` e `getSecond()`. E' fornita di un metodo che, data un'altra coppia restituisce la coppia somma.



Programmazione ad Oggetti - © S. Cicerone, G. Di Stefano

Incapsulamento: *accesso diretto agli attributi negato*

Data la definizione di Robot:



Il seguente codice è errato:

```
//tentativo di accesso diretto
//allo stato del Robot

//creazione di un oggetto Robot
var rob: Robot := Robot.New();

//pos è [0,0]
var pos: Posizione :=
    Posizione.New();

//accesso diretto
rob.posizione = pos; //Errore!

rob.setPosizione(pos); //OK
```

Programmazione ad Oggetti - © S. Cicerone, G. Di Stefano

Esercizio

- Sulla base della classe Robot, riportata a fianco, e della classe Pair, riprogettare le classi Punto, Cerchio, Data e Ora specificando attributi e metodi.

