



UNIVERSITÀ DEGLI STUDI DI SALERNO
DIPARTIMENTO DI INFORMATICA
DIPARTIMENTO DI ECCELLENZA

Università degli Studi di Salerno

Dipartimento di Informatica

Programmazione ad Oggetti

a.a. 2023-2024

Paradigma di Programmazione ad Oggetti

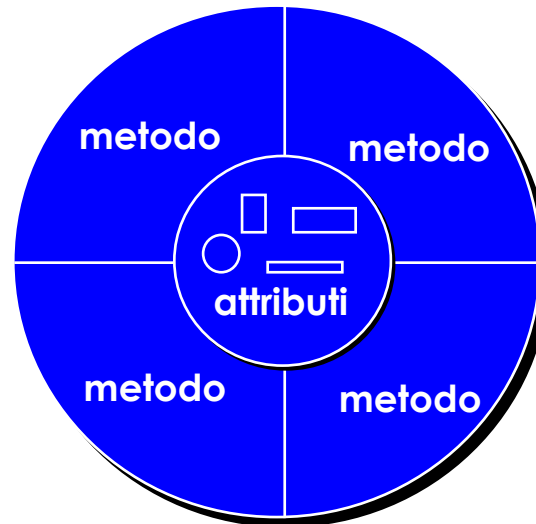
Docente: Prof. Massimo Ficco

E-mail: *mficco@unisa.it*

Tipo Dati Astratto

ASTRAZIONE SUI DATI

Il tipo di dati astratto (ADT)



- Una Classe implementa un tipo di dato astratto.
- Una classe ha un nome, e contiene attributi e metodi

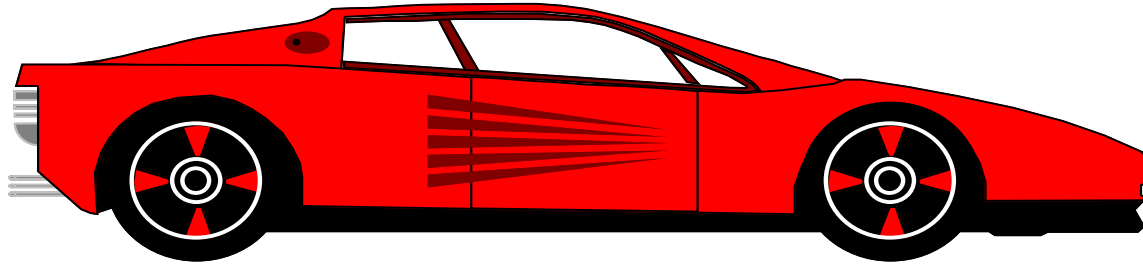


Oggetto

- Un oggetto è una istanza di una classe (a volte di più di una). Un oggetto occupa memoria, la sua classe definisce come sono organizzati i dati in questa memoria. Ogni oggetto possiede tutti gli attributi definiti nella classe, ed essi hanno un valore, che può mutare durante l'esecuzione del programma come quello di qualsiasi variabile. Se il paradigma della OOP è applicato in modo rigido, gli attributi di un oggetto vengono manipolati solo da metodi invocati su quello stesso oggetto.
- Sintatticamente, i metodi di una classe vengono invocati su un particolare oggetto, e ricevono come parametro implicito l'oggetto su cui sono stati invocati. Questo parametro normalmente può essere referenziato esplicitamente; per esempio, in C++, in Java, e in C# si usa la parola chiave `this`, mentre in Smalltalk, in Objective-C, e in Ruby si usa `self`.
- Alcuni linguaggi forniscono un supporto per l'inizializzazione automatica di un oggetto, con uno speciale metodo detto **costruttore**. Allo stesso modo, la fine della vita di un oggetto è gestita con un metodo detto **distruttore**



ESEMPIO di Classe: AUTOMOBILE



Funzioni

- Avviati
- Fermati
- Accelera
- ...

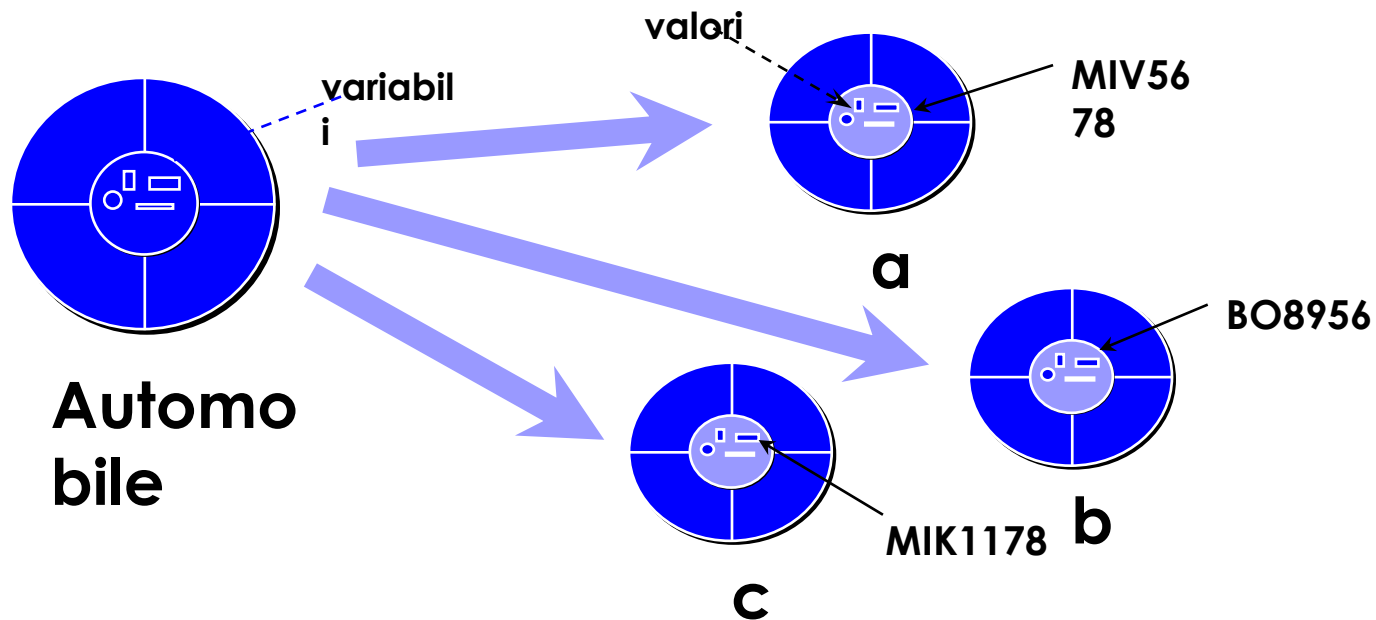
Dati:

- Targa
- Colore
- Velocità
- Livello benzina
-
-



OGGETTI

Un oggetto è una istanza (“esemplare”) di una classe, che viene creato (“istanziato”) dinamica-mente



Due esemplari della stessa classe sono distinguibili soltanto per il loro stato (il valore dei loro campi), mentre il comportamento è sempre identico



OGGETTI (Cont.)

Modellano le entità del dominio di applicazione (proprietà ed interrelazioni)

Versione informatica degli oggetti "reali"

Dotati di una loro propria "individualità"

Capaci di interagire per scambio di messaggi

Caratterizzati da proprietà

- dati: ne descrivono lo “stato”
- funzioni: ne descrivono il “comportamento”

Un msg modifica stato e attiva comportamenti

Un oggetto è una coppia [stato,funzioni]



INTERAZIONI TRA OGGETTI

Un msg modifica stato e attiva comportamenti

Gli oggetti possono comunicare e interagire mediante scambio di messaggi attraverso le loro interfacce pubbliche (stato o funzioni)

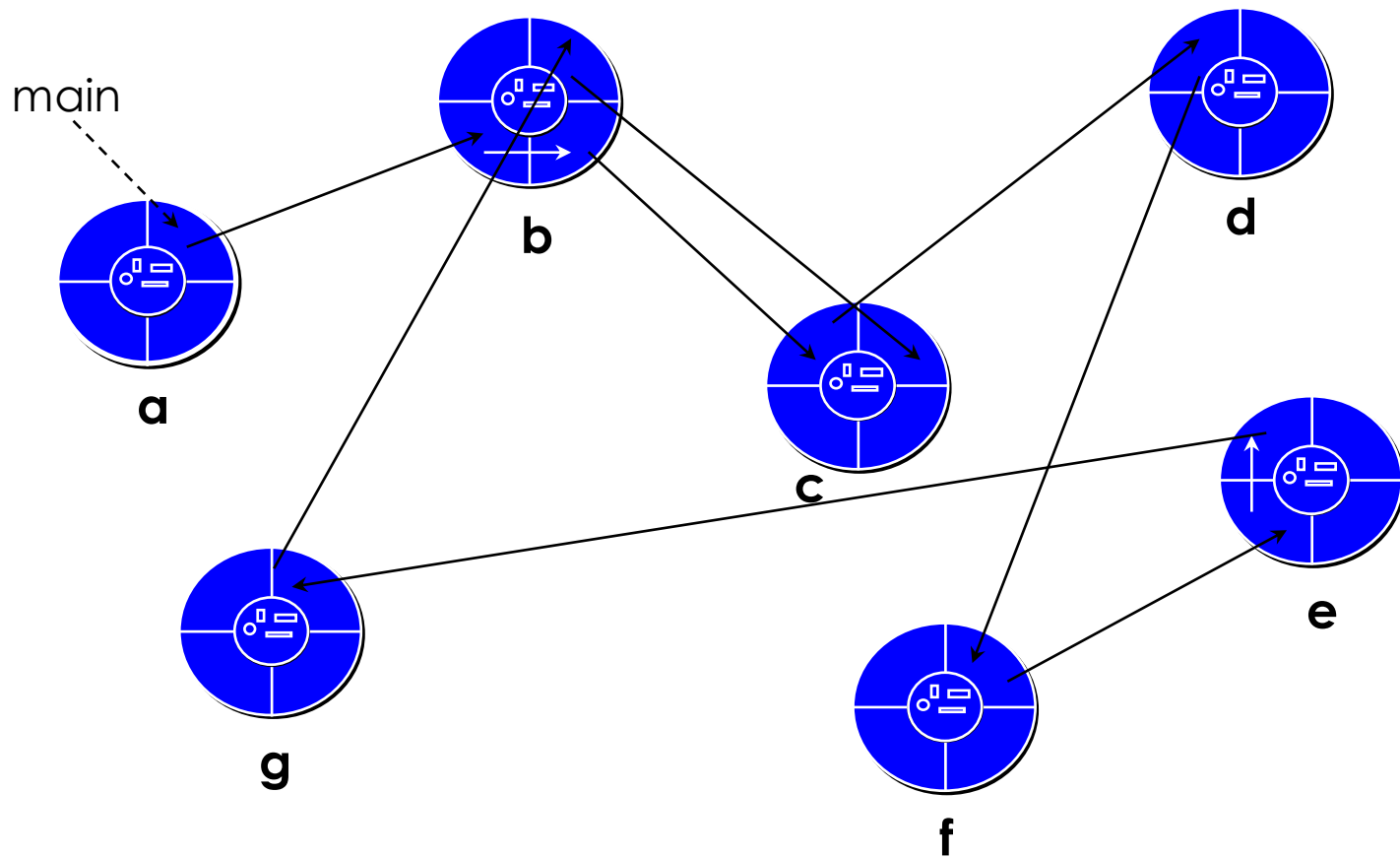
Per mezzo di un messaggio un oggetto può chiedere un'informazione a un altro oggetto, causarne un cambiamento di stato, oppure delegargli un'attività

Un messaggio in arrivo viene trattato dal metodo omonimo del ricettore, il quale "si attiva" per rispondere, per cambiare di stato, oppure per intraprendere un'attività



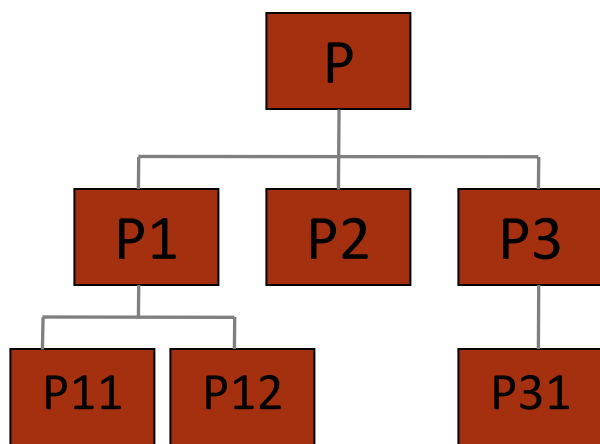
UN SISTEMA A OGGETTI

chiamata di un metodo di b



Metodologie di progettazione

Metodologia discendente o “**top-down**”: basata sulla decomposizione funzionale nella definizione del sistema software
=> individuazione delle funzionalità e raffinamenti successivi



Metodologia ascendente o “**bottom-up**”: basata su individuazione di entità del sistema (classi e/o oggetti), delle loro proprietà e delle interrelazioni tra di esse.

Il sistema viene costruito assemblando componenti con un approccio “dal basso verso l’alto”.

La modellazione a oggetti è di tipo bottom-up



Tecniche di programmazione ad oggetti

Si parla di **programmazione con oggetti** con riferimento a tecniche di programmazione basate sul concetto di oggetto (**dati+operazioni**)

Si parla di **programmazione basata sugli oggetti** (**object-based programming**) con riferimento alle tecniche di programmazione basate sui concetti di:

- Tipo di dati astratto o Classe (tipo)
- Oggetto (istanza di un tipo di dati astratto)

Si parla di **programmazione orientata agli oggetti** (**object-oriented programming, OOP**) con riferimento alle tecniche basate sui concetti:

- Classe
- Oggetto
- Ereditarietà
- Polimorfismo



Linguaggi ad oggetti (2/2)

Esistono dunque linguaggi ad oggetti:

Non tipizzati

- Es.: Smalltalk
- E' possibile definire oggetti senza dichiarare il loro tipo
- In tali linguaggi, gli oggetti sono entità che incapsulano una struttura dati nelle operazioni possibili su di essa

Tipizzati

- Es.: C++, Java
- E' possibile definire tipi di dati astratti e istanziarli
- Gli oggetti devono appartenere ad un tipo (astratto)
- In tali linguaggi, una **classe** è una implementazione di un tipo di dati astratto. Un **oggetto** è una istanza di una classe



Il linguaggio C++

C++ è un linguaggio di programmazione **general-purpose** che supporta:

- la *programmazione procedurale* (è un C “migliore”)
- la *programmazione orientata agli oggetti*
- la *programmazione generica*

C++ è quindi un **linguaggio ibrido**, nel senso che supporta più paradigmi di programmazione

