

## Classi e oggetti

- Cos'è una classe?
- Cos'è un oggetto?
- Che differenza c'è tra classe e oggetto?
- Cos'è l'identificatore di un oggetto?
- Che differenza c'è tra le classi in Java e in C++?
- In che senso "Java non ha i puntatori"?
- Com'è definita la firma di un metodo?
- Qual è il ruolo del package?
- Cos'è la variabile "this"?
- Come faccio a stampare un oggetto?

## Memoria

- Quando (e come) viene usato lo stack?
- Quando (e come) viene usata la heap?
- Ci sono variabili che non stanno né in heap né in stack?
- Cos'è un memory leak?
- Cos'è il garbage collector? Come fa a sapere cosa può essere rimosso?

## Classi e interfacce

- Cos'è una classe astratta? Per cosa si usa?
- Cos'è una interfaccia? Per cosa si usa?
- Che differenza c'è tra classi astratte e interfacce?

## Static

- Che differenza c'è tra le variabili e metodi statici e quelli "normali"?
- Perché l'uso di variabili statiche va limitato? Quando è lecito usarle?

## Visibilità

- Quali sono i livelli di visibilità previsti in Java e qual è il loro significato?
- Che differenza c'è tra non dichiarare il livello di visibilità e dichiarare protected?

## Ereditarietà

- Che differenze ci sono per quanto riguarda l'ereditarietà tra classi astratte e interfacce?
- Cos'è una is-a? e una has-a?
- Che differenza c'è tra overriding e overloading?
- Cosa si intende per dynamic binding?
- Quando in Java non si ha dynamic binding?
- Come viene deciso quale metodo eseguire in presenza di polimorfismo?
- Che relazione c'è tra costruttore (e distruttore) ed ereditarietà?
- Cos'è la variabile "super"?
- Cosa si intende per "upcast" e quando è lecito?
- Cosa si intende per "downcast" e quando è lecito?
- Cos'è e a cosa serve l'operatore instanceof?
- A cosa serve l'annotazione @Override?

## Equals

- Come si scrive una equals corretta?
- Qual è il ruolo della hashCode?

## Comparazione ai fini dell'ordinamento

- Come si scrive una compareTo corretta?
- Che differenza c'è tra Comparable e Comparator?

## Collections

- Perché quasi sempre conviene usare collections invece che arrays?
- Che differenza c'è tra liste e set?
- Quando una collezione è ordinabile?
- Come posso mescolare/ordinare una Collection?
- Perché la definizione di equals e hashCode è importante per le classi i cui oggetti sono inseriti in una Collection?
- Come posso scorrere gli elementi di una collezione?
- Come si usa un iteratore?
- Che differenza c'è tra usare un iteratore e usare un ciclo for generalizzato (for Tipo x : Collection) ?

## Gestione degli errori – terminazione dei programmi.

- Cos'è e a cosa serve la assert?
- Come si usa il costrutto try/catch?
- A che serve la clausola finally?
- Come faccio a fare terminare un programma?