Dipartimento di Informatica Sistemistica e Comunicazione
Corso di Laurea in Informatica

Programmazione 2

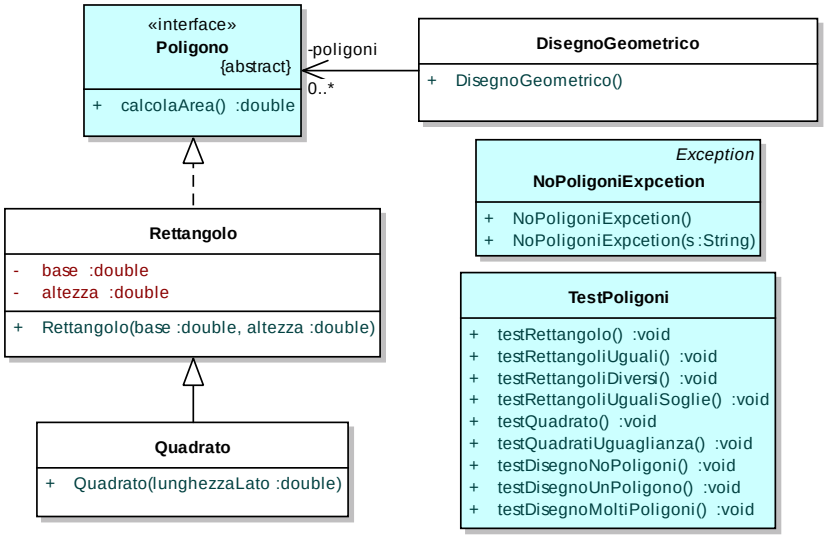
12 Giugno 2015 – Secondo compitino

Testo parte di pratica

Si realizzino le classi che modellano dei poligoni e delle figure geometriche come da diagramma UML e specifiche che seguono. Si provino le classi utilizzando i test forniti nella classe `TestPoligoni`.

Nota: Gli elaborati che non superino almeno 4 test fra quelli dati saranno considerati insufficienti.

(**Suggerimento: si eviti di eseguire i test solo alla fine del lavoro, quando ormai sarebbe tardi per apportare correzioni. Piuttosto si eseguano i test man mano che si procede con l'implementazione, per verificare incrementalmente il lavoro via via fatto.)**



Le classi in azzurro sono fornite e non sono quindi descritte nel seguito. Per la comprensione di queste classi si rimanda all'implementazione.

Gli attributi di tutte le classi che devono essere realizzate **sono immutabili** e **non accessibili** dall'esterno.

Infine, **il diagramma non specifica i metodi d'istanza delle classi da implementare**. La descrizione di tali metodi è riportata di seguito. Tutti i metodi da implementare hanno visibilità pubblica.

Classi `Rettangolo` e `Quadrato`:

- Sono poligoni. La classe `Rettangolo` rappresenta un rettangolo caratterizzato dalle misure di base e altezza da impostare con il costruttore. La classe `Quadrato` rappresenta un quadrato che è un particolare rettangolo caratterizzato dalla sola misura di un alto da impostare con il costruttore.
- Poiché poligoni, è possibile invocare il metodo `calcolaArea():double` sia su oggetti di tipo `Rettangolo` sia su oggetti di tipo `Quadrato`. Il metodo `calcolaArea():double` deve ritornare il valore dell'area del poligono.
- Infine, è possibile confrontare rettangoli e quadrati eseguendo il metodo `equals(o:Object):boolean` (override di quello ereditato dalla classe `Object`) che confronta due poligoni dello stesso tipo (nel caso, due rettangoli o due quadrati) secondo la seguente specifica: i poligoni sono uguali se hanno la medesima area o se le rispettive aree differiscono per meno di 0.001. (Suggerimento: si ricorda l'esistenza del metodo `Math.abs(val:double):double`, che restituisce il valore assoluto del numero passato come parametro).

Classe `DisegnoGeometrico`:

- Rappresenta un disegno geometrico che include uno o più poligoni. L'associazione `poligoni` viene gestita usando una collezione di tipo `ArrayList`.
- Il metodo `aggiungiPoligono(poligono:Poligono):void` aggiunge al disegno geometrico il poligono passato come parametro.
- Il metodo `calcolaAreaTotale():double` restituisce il valore che si ottiene sommando le aree di tutti i poligoni inclusi nel disegno geometrico. Se nessun poligono è presente, il metodo solleva l'eccezione `NoPoligoniException`.