

Esercizi sulla creazione di classi

(Fondamenti di Informatica – Emilio Di Giacomo)

Esercizio 1 Definire una classe di nome **Complesso** per rappresentare numeri complessi. Ogni istanza della classe è un numero complesso ed è perciò rappresentato per mezzo di due numeri double **re** e **im** che rappresentano rispettivamente la parte reale e la parte immaginaria del numero.

La classe deve essere dotata dei seguenti costruttori e metodi:

- /* crea un oggetto Complesso con parte reale re e parte immaginaria im */
public Complesso (double re, double im)
- /* restituisce il modulo del numero complesso rappresentato dall'oggetto ricevente (cioè quello su cui è invocato il metodo) */
public double modulo ()
- /* restituisce un oggetto Complesso che è il coniugato del numero complesso rappresentato dall'oggetto ricevente (cioè quello su cui è invocato il metodo)*/
public Complesso coniugato ()
- /* restituisce un oggetto Complesso che è pari alla somma tra il numero complesso rappresentato dall'oggetto ricevente (cioè quello su cui è invocato il metodo) e il numero complesso c passato per parametro */
public Complesso somma (Complesso c)
- /* restituisce un oggetto Complesso che è pari al prodotto tra il numero complesso rappresentato dall'oggetto ricevente (cioè quello su cui è invocato il metodo) e il numero complesso c passato per parametro */
public Complesso prodotto (Complesso c)
- /* restituisce una descrizione testuale del numero complesso rappresentato dall'oggetto ricevente (cioè quello su cui è invocato il metodo) */
public String toString ()

Si ricorda che:

- il coniugato del numero complesso $re + i \cdot im$ è il numero complesso $re - i \cdot im$;
- il modulo di $re + i \cdot im$ è il numero reale $\sqrt{re^2 + im^2}$
- la somma di $re_1 + i \cdot im_1$ e $re_2 + i \cdot im_2$ è pari a $(re_1 + re_2) + i \cdot (im_1 + im_2)$
- il prodotto di $re_1 + i \cdot im_1$ e $re_2 + i \cdot im_2$ è pari a $(re_1 \cdot re_2 - im_1 \cdot im_2) + i \cdot (im_1 \cdot re_2 + re_1 \cdot im_2)$

Una volta definita la classe **Complesso**, scrivere una classe **ProvaComplesso** che ne verifica il corretto funzionamento. La classe **ProvaComplesso** ha il solo metodo **main**, fa inserire all'utente due numeri complessi $re_1 + i \cdot im_1$ e $re_2 + i \cdot im_2$ e mostra all'utente i coniugati dei due numeri inseriti, i loro moduli, la loro somma e il loro prodotto.

Esercizio 2 Utilizzando la classe **Complesso** definita nell'esercizio precedente si definisca una classe **EquazioneDiSecondoGrado** che rappresenta equazioni di secondo grado nella forma $ax^2 + bx + c = 0$. La classe dovrà avere i seguenti costruttori e metodi:

- /* crea un oggetto EquazioneDiSecondoGrado con coefficienti a, b e c */
public EquazioneDiSecondoGrado (double a, double b, double c)
- /* restituisce la prima radice dell'equazione */
public Complesso primaRadice ()
- /* restituisce la seconda radice dell'equazione */
public Complesso secondaRadice ()

Per calcolare le due radici bisogna prima determinare se le soluzioni sono reali o meno cioè guardare il segno del discriminante $\Delta = b^2 - 4ac$. Se $\Delta \geq 0$ le soluzioni sono reali (cioè numeri complessi con parte immaginaria pari a 0) e sono date da:

$$x_1 = \frac{-b + \sqrt{b^2 - 4ac}}{2a} \quad x_2 = \frac{-b - \sqrt{b^2 - 4ac}}{2a};$$

Se $\Delta < 0$ le soluzioni sono complesse e sono date da:

$$x_1 = -\frac{b}{2a} + i \cdot \frac{\sqrt{4ac - b^2}}{2a} \quad x_2 = -\frac{b}{2a} - i \cdot \frac{\sqrt{4ac - b^2}}{2a}.$$

Una volta definita la classe **EquazioneDiSecondoGrado**, scrivere una classe **ProvaEquazioneDiSecondoGrado** che ne verifica il corretto funzionamento. La classe **ProvaEquazioneDiSecondoGrado** avrà il solo metodo **main**, che fa inserire all'utente i coefficienti di un'equazione di secondo grado e mostra all'utente le sue due radici.

Esercizio 3 Definire una classe di nome **Distributore**, le cui istanze modellano distributori di carburante di una certa compagnia. Precisamente, un oggetto **Distributore** deve avere i seguenti campi:

- **Nome** del gestore (di tipo **String**),
- **Cognome** del gestore (di tipo **String**),
- **Indirizzo** (di tipo **String**),
- **Scorta** (di tipo **double**) indica la quantità di carburante disponibile per la vendita.

La classe **Distributore** ha inoltre una variabile statica, di nome **prezzoCarburante**, che indica il prezzo al litro del carburante.

La classe deve avere i due costruttori seguenti:

- /* crea un oggetto Distributore con gestore e indirizzo specificati; la scorta è fissata a 0 dal costruttore */
public Distributore (String nomeGestore, String cognomeGestore, String indirizzo)
- /* crea un oggetto Distributore con gestore, indirizzo e scorta specificati */
public Distributore (String nomeGestore, String cognomeGestore, String indirizzo, **double** scorta)

La classe deve inoltre avere i seguenti metodi di istanza:

- /* effettua un rifornimento di carburante. Il parametro litri indica i litri prelevati che vanno quindi sottratti alla scorta. Il metodo restituisce la somma da pagare per il rifornimento effettuato */
`public double rifornisci (double litri)`
- /* aggiunge alla scorta la quantità di litri passata come parametro */
`public void ricaricaScorta (double litri)`
- /* restituisce una descrizione completa del distributore */
`public String toString ()`

La classe ha infine il seguente metodo statico:

- /* modifica il valore del prezzo al litro del carburante con la nuova cifra indicata */
`public static void setPrezzo (double prezzo)`

Una volta definita la classe **Distributore**, scrivere una classe **ProvaDistributore** che ne verifica il corretto funzionamento. La classe **ProvaDistributore** avrà il solo metodo **main**, che svolge le seguenti azioni:

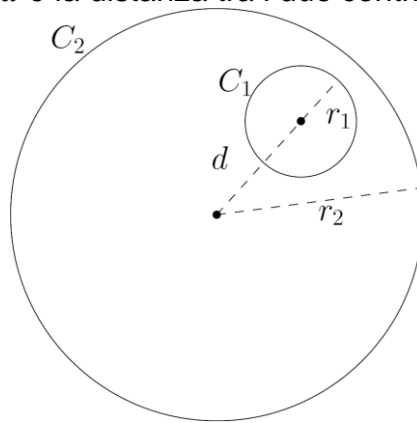
- Fissa il prezzo al litro del carburante a 1,369 €.
- Crea un oggetto **Distributore** facendo impostare all'utente tutti i dati ad esso relativo, compresa la scorta iniziale.
- Visualizza all'utente una descrizione completa del distributore creato.
- Effettua un rifornimento chiedendo all'utente i litri da rifornire.
- Visualizza all'utente la somma da pagare per il rifornimento effettuato.
- Visualizza all'utente la nuova descrizione del distributore.
- Effettua una ricarica della scorta di carburante chiedendo all'utente i litri da aggiungere alla scorta.
- Visualizza all'utente la nuova descrizione del distributore.

Esercizio 4 Definire una classe di nome **Circonferenza** le cui istanze modellano cerchi nel piano descritti da un'equazione nella forma $x^2 + y^2 + ax + by + c = 0$. La classe deve essere dotata dei seguenti costruttori e metodi:

- /* crea un oggetto Circonferenza con la cui equazione ha coefficienti a, b, e c */
`public Circonferenza (double a, double b, double c)`
- /* restituisce il centro della circonferenza rappresentata dall'oggetto ricevente (cioè quello su cui è invocato il metodo) */
`public Punto centro ()`
- /* restituisce il raggio della circonferenza rappresentata dall'oggetto ricevente (cioè quello su cui è invocato il metodo) */
`public double raggio ()`
- /* restituisce true se la circonferenza rappresentata dall'oggetto ricevente (cioè quello su cui è invocato il metodo) contiene completamente la circonferenza c passata come parametro */

public boolean contiene (Circonferenza c)

Si ricorda che data una circonferenza di equazione $x^2 + y^2 + ax + by + c = 0$ le coordinate del centro sono $x_c = -\frac{a}{2}$ e $y_c = -\frac{b}{2}$, mentre il raggio è dato da $r = \sqrt{\frac{a^2}{4} + \frac{b^2}{4} - c}$. Si ricorda inoltre che una circonferenza C_1 di raggio r_1 è contenuta in una circonferenza C_2 di raggio r_2 se $d + r_1$ è minore di r_2 , dove d è la distanza tra i due centri. (vedi Figura).



La classe **Punto** (da utilizzare per restituire il centro della circonferenza) è data ed è la stessa già usata nell'esercitazione 1.

Una volta definita la classe **Circonferenza**, scrivere una classe **ProvaCirconferenza** che ne verifica il corretto funzionamento. La classe **ProvaCirconferenza** avrà il solo metodo **main**, che svolge le seguenti azioni:

- Fa inserire i coefficienti dell'equazione di due circonferenze C_1 e C_2 ;
- Visualizza centro e raggio delle due circonferenze;
- Dice all'utente se la circonferenza C_1 è contenuta in C_2 .

Esercizio 5 Definire una classe di nome **PoligonoRegolare**, le cui istanze modellano poligoni regolari.

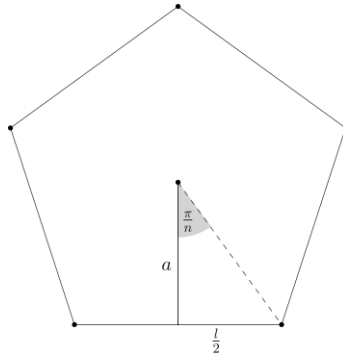
La classe deve avere il seguente costruttore:

- /* crea un oggetto PoligonoRegolare con n lati di lunghezza l */
public PoligonoRegolare (*int* n, *double* l)

La classe deve inoltre avere i seguenti metodi di istanza:

- /* cambia la misura del lato del poligono */
public void cambiaDimensione (*double* l)
- /* restituisce l'apotema del Poligono */
public double apotema ()
- /* restituisce il perimetro del Poligono */
public double perimetro ()
- /* restituisce l'area del poligono */
public double area ()

Si ricorda che l'apotema a di un poligono regolare è il segmento che unisce il centro del poligono al punto medio di un lato (vedi figura) e che può essere calcolato come $a = \frac{l}{2 \tan \frac{\pi}{n}}$, dove l è la lunghezza del lato e n è il numero di lati. Si ricorda inoltre che l'area di un poligono regolare è pari al prodotto del perimetro per l'apotema diviso 2.



Una volta definita la classe **PoligonoRegolare**, scrivere una classe **ProvaPoligonoRegolare** che ne verifica il corretto funzionamento. La classe **ProvaPoligonoRegolare** avrà il solo metodo **main**, che svolge le seguenti azioni:

- Fa creare un poligono regolare all'utente chiedendogli quanti lati deve avere il poligono e quanto questi devono essere lunghi.
- Visualizza all'utente apotema, perimetro e area del poligono creato.
- Chiede all'utente di cambiare la misura del lato del poligono.
- Visualizza all'utente apotema, perimetro e area del poligono modificato.