

1. Definite la classe **Complex** per effettuare operazioni matematiche sui numeri complessi. Scrivete un programma di prova per verificare l'adeguatezza della vostra classe.

I numeri complessi hanno la seguente forma:

Re + i *Im

dove **Re** è la parte reale del numero, **Im** è la parte immaginaria e **i** indica la radice quadrata di -1.

Rappresentate i dati private della classe tramite variabili a virgola mobile. Scrivete un costruttore che inizializzi ogni oggetto della classe quando è dichiarato. Il costruttore deve contenere dei valori di default, nel caso che non siano forniti inizializzatori. Ognuna delle quattro operazioni seguenti dovrebbe essere svolta da una funzione membro diversa:

- a) Addizione di due numeri complessi: parti reali e parti immaginarie vanno sommate separatamente.
- b) Sottrazione di due numeri complessi: la parte reale dell'operando destro è sottratta dalla parte reale dell'operando di sinistra, e lo stesso vale per le parti immaginarie.
- c) Visualizzazione di numeri complessi nella forma **(a, b)** dove **a** è la parte reale e **b** quella immaginaria

2. Definite la classe **Rectangle**. Questa classe ha gli attributi **length** e **width**, ognuno dei quali per default vale 1. Ha due funzioni membro che calcolano il perimetro e l'area del rettangolo e funzioni **set** e **get** per **length** e **width**. Le funzioni **set** devono verificare che **length** e **width** siano numeri a virgola mobile compresi fra 0.0 e 20.0.

3. Progettate la classe "String". La classe deve fondamentalmente:

- a) contenere una stringa (array di char) di cui fisserete la lunghezza massima,
- b) permettere le operazioni più comuni (definite, tra gli altri, un metodo che restituisce la lunghezza della stringa "len()" ed un metodo per concatenare 2 stringhe "append()")
- c) avere un metodo che restituisce il suo contenuto in formato standard "C" (array di char con '\0' alla fine).

Fate attenzione a che i metodi siano tutti intrinsecamente sicuri (non deve essere possibile scrivere su zone di memoria non riservate come invece accade per le funzioni della libreria "string.h"). Fornite almeno la definizione della classe (il file "String.h"), l'implementazione dei metodi len() e append(...) e una descrizione funzionale degli altri metodi che pensate di includere.

4. Includete la stampa di un messaggio nei costruttori e nel distruttore delle classi sviluppate in precedenza e verificate, con dei semplici programmi di test, quando questi vengono invocati nel caso di variabili statiche, globali e locali.
5. Rendete **const** tutti i metodi che ritenete opportuni nelle classi sviluppate negli esercizi 1-3 e verificate il corretto funzionamento con dei programmi di test che usano oggetti **const**.