

Esercizio – classe Razionale

Estendere la classe Razionale mostrata a lezione (disponibile on-line all'indirizzo <http://elly.smfi.unipr.it/2017/mod/folder/view.php?id=1989>) in modo che possa trattare anche i numeri interi come caso particolare di numeri razionali.

In particolare la classe estesa dovrà fornire, in aggiunta a quanto già presente:

- un costruttore con un parametro intero n che costruisce il numero razionale $n/1$;
- operazioni di lettura e stampa (con overloading degli operatori $<<$ e $>>$) di un numero intero da/su stream: nell'operazione di lettura di un razionale, se lo stream di input contiene un numero intero n seguito da “a capo” (invece che da '/'), il valore memorizzato nel razionale è $n/1$; nell'operazione di stampa di un razionale, se la frazione può essere ridotta ad un intero n , allora viene stampato n ;
- operazioni di somma, prodotto e confronto (con overloading rispettivamente degli operatori $+$, $*$ e $<$), tra due numeri razionali e tra un numero razionale ed un numero intero.

Inoltre, tutte le frazioni devono essere ridotte ai minimi termini.

Note di implementazione.

Realizzare (come funzioni proprie private) le seguenti funzioni:

- funzione `mcd(x, y)`, di tipo `int`: calcola il massimo comune divisore tra due interi x e y ;
- funzione `semplifica()`, di tipo `void`: riduce ai minimi termini la frazione su cui è invocata (utilizza la funzione `mcd`).

Main di prova.

La classe realizzata deve essere inserita nel main di prova messo a disposizione su 'elly' insieme a questo testo (file `razionale_main.cpp`). L'esecuzione del main, completato con la classe Razionale, deve dare i risultati riportati di seguito al main stesso. Non è ammesso apportare modifiche al main.

N.B. E' vietato: usare dichiarazioni friend, usare goto, usare variabili globali non motivate, definire public tutti i campi di una classe, usare ereditarietà a sproposito.