

# Esercizi - C++: Tipi di dato strutturato

---

## 1. I razionali

Scrivere una classe per la rappresentazione dei numeri naturali che supporti tutte le operazioni algebriche su di essi e gli operatori `<<` e `>>`.

## 2. Implementare gli `shared_ptr`

Implementare una versione semplificata della classe template `shared_ptr` che fornisca i seguenti metodi:

- i costruttori, `shared_ptr(T*)` e `shared_ptr(const shared_ptr&)`
- il distruttore, `~shared_ptr()`
- gli operatori di de-referenziazione, `operator*` e `operator-->`
- l'operatore di copia, `operator=`
- l'operatore di cast verso il tipo `bool`, `operator bool`
- il metodo `unsigned int use_count() const` che restituisce il numero di riferimenti allo *shared pointer*

## 3. Il cambiamonete

Scrivere una classe `Cambiamonete` che fornisca le funzionalità richieste dall'esercizio 4 della sezione 3.

## 4. Gli studenti

Implementare una classe per rappresentare uno studente. Tale classe dovrà caratterizzare gli studenti con il loro nome e cognome, il loro numero di matricola, gli esami sostenuti e, per ciascun esame superato, il voto conseguito.

## 5. Ordinare un array

Scrivere una funzione *template* che, dato un array di valori di un tipo di dato strutturato, lo ordini.

Usare la funzione per ordinare:

1. un array di stringhe lessicograficamente
2. un array di `studente` (vedi Es. 2) rispetto al numero di matricola
3. un array di `studente` (vedi Es. 2) rispetto alla media dei voti conseguiti