

Esercizi di programmazione*

1 Lezione 10

Suggerimento: per la lettura e scrittura di stringhe, numeri interi e reali, si faccia riferimento alla libreria `IOCREMA.H`

Esercizio 1 Si scriva un programma che definisce il tipo `rational` per rappresentare i numeri razionali, costituito da due campi `num` e `den` che conterranno il numeratore e il denominatore del numero stesso. Si definiscano funzioni che ricevono due numeri razionali e restituiscono la loro somma, differenza, prodotto, rapporto.

Nota: Questo esercizio si può complicare, chiedendo che, al termine delle operazioni, `num` e `den` siano semplificati sino a diventare primi fra loro. Per farlo, occorre calcolarne il massimo comun divisore, per esempio con l'algoritmo di Euclide.

Esercizio 2 Si scriva un programma che definisce il tipo `complex` per rappresentare i numeri complessi, costituito da due campi `r` e `i` che conterranno la parte reale e immaginaria del numero stesso. Si definiscano funzioni che ricevono due numeri complessi e restituiscono la loro somma, differenza, prodotto.

Esercizio 3 Si scriva un programma che definisce il tipo `data` per rappresentare le date, intese come terne giorno-mese-anno. Per il mese, si definisca un tipo enumerativo *ad hoc*. Si definiscano:

- una funzione che riceve una data e restituisce la posizione del giorno lungo l'anno (da 1 a 366)
- una funzione che riceve due date e restituisce -1 se la prima data precede la seconda, 0 se coincidono, $+1$ se la prima data segue la seconda

Esercizio 4 Si scriva un programma che definisce il tipo `time` contenente tre campi `ora`, `minuto` e `secondo`. Si definisca una funzione che riceve un valore `long secondi_totali`, lo interpreta come il numero di secondi trascorsi dalla mezzanotte e lo traduce in una struttura di tipo `time`.

Esercizio 5 Si scriva un programma che definisce il tipo `colore` contenente tre campi `R`, `G`, e `B`, corrispondenti all'intensità luminosa nel campo del rosso, del verde e del blu (valori compresi fra 0 e 255). Si definisca una funzione `Schiarisce()` che riceve un colore e restituisce un colore più chiaro, dividendo per 0.7 tutti i valori e arrotondandoli all'intero più vicino compreso fra 0 e 255 .

*tratti o ispirati dal testo di K.N. King

Esercizio 6 Si scriva un programma che definisce il tipo `point` contenente due campi interi `x` e `y` per rappresentare un punto su un piano e il tipo `rectangle` che rappresenta un rettangolo attraverso due campi `nordovest` e `sudest` che rappresentano i due punti estremi in alto a sinistra e in basso a destra. Si definiscano funzioni che:

- dati due punti, restituisca il rettangolo da loro individuato
- dato un rettangolo, ne restituisca l'area
- dato un rettangolo e un punto, restituisca un valore `boolean` (tipo enumerativo da definire) che indichi se il punto sta nel rettangolo (bordi compresi) o fuori

Come cambierebbero tali funzioni se i punti avessero coordinate reali?

Esercizio 7 Si definiscano due tipi enumerativi: `pezzo`, con valori `RE`, `REGINA`, `TORRE`, `ALFIERE`, `CAVALLO`, `PEDONE`, `VUOTO` e `colore`, con valori `BIANCO`, `NERO`, `VUOTO`. Si definisca poi una struttura `casella`, che rappresenta una casella del gioco degli scacchi, componendo i due tipi precedenti nonché la riga e la colonna. Infine, si definisca un tipo `scacchiera` come matrice quadrata di caselle. Si scriva un programma con una funzione che inizializza le posizioni dei pezzi su una scacchiera e un'altra funzione che stampa a video la scacchiera con le posizioni correnti.