



ALMA MATER STUDIORUM  
UNIVERSITÀ DI BOLOGNA

# Fondamenti di Informatica T1 - Laboratorio

## Simulazione Esame

# Consegna Esercizio (importante)

**Prima di cominciare:** si scarichino i file di testo dallo **StartKit.zip** dopo aver opportunamente unzippato la cartella e i file contenuti in essa.

## Avvertenze per la consegna:

1) nominare TUTTI i file sorgenti **come richiesto nel testo del compito**, apporre all'inizio di **ogni file** sorgente un commento contenente i propri dati (**cognome, nome, numero di matricola**).

2) Al termine, **consegnare tutti i file sorgenti** (i due file .c e il file .h) **ed i file di testo contenuti nello StartKit.**

**NON CONSEGNARE CARTELLE, PROGETTI O .ZIP**

3) Rispettare le specifiche del testo, in particolare inserire le funzioni con i nomi, i parametri di ingresso e i parametri di uscita **esattamente come indicato nel testo.** Chi non rispetta le specifiche sarà opportunamente penalizzato.

4) **ALL'ESAME SARANNO CORRETTI SOLO gli elaborati SENZA errori di compilazione.**



# Simulazione Esame - Testo

L'azienda Peppers&Peppers utilizza un sistema informatizzato per gestire le vendite di piante di peperoncino piccante.

In un file di testo chiamato "*prodotti.txt*" sono memorizzate le informazioni sulle piante disponibili per la vendita; ogni riga corrisponde ad una varietà di pianta ed è formata dai seguenti campi:

**codice nome valore quantita prezzo**

dove *codice* è una stringa di 6 caratteri senza spazi che identifica in maniera univoca la varietà di pianta, *nome* è una stringa senza spazi (massimo 40 caratteri) che rappresenta il nome della varietà, *valore* è un intero senza segno che rappresenta la piccantezza della varietà di peperoncino misurata in unità Scoville nell'intervallo  $[0...16000000]$ , *quantita* è un intero che rappresenta la quantità di piante di quella varietà disponibili alla vendita, *prezzo* è un valore reale a precisione singola che rappresenta il prezzo di una singola pianta.



# Simulazione Esame - Testo

Un altro file di testo denominato "*acquisti.txt*" contiene le informazioni relative agli acquisti, ogni riga del file è composta dai seguenti campi:

**cf\_cliente codice quantita**

dove *cf\_cliente* è il codice fiscale del cliente (16 caratteri senza spazi), *codice* è una stringa di 6 caratteri senza spazi che identifica la varietà di pianta e *quantita* è un numero intero che rappresenta il numero di piante acquistate dal cliente. Ogni acquisto è identificato in maniera univoca dalla coppia *<cf\_cliente, codice>*.

Tutti i codici nella lista degli acquisti sono sicuramente contenuti nella lista dei prodotti.



# Esercizio 1

## Esercizio 1 - Lettura da file (piante .h/piante .c)

Si definisca un'opportuna struttura dati Peperoncino che rappresenti una pianta di peperoncino.

Il candidato realizzi poi la funzione:

**int leggiProdotti (char\* nomefile, Peperoncino vp[], int\* dim)**

che, ricevuto come parametro di ingresso il nome di un file, provveda ad aprirlo e ne legga il contenuto, ovvero la lista di varietà di peperoncino disponibili. Non è noto a priori il numero esatto di righe del file, per cui il vettore è allocato nel *main* con una dimensione massima MAX\_DIM (ad esempio 100) e passato alla funzione tramite il parametro **vp**. La funzione restituisce la dimensione logica tramite il parametro **dim**: come valore di ritorno restituisce 0 in caso di corretta lettura, -1 in caso di errore in apertura del file.

Si realizzi inoltre la procedura:

**void stampaProdotto (Peperoncino p)**

che stampi a video (in maniera leggibile – usare spazi, separatori, fine linea, etc.) le informazioni relative alla varietà di peperoncino passata in ingresso.



# Esercizio 1

Procedendo in maniera simile al punto precedente, definire un'opportuna struttura dati **Acquisto** per rappresentare un acquisto fatto da un cliente. Realizzare quindi la funzione:

**int leggiAcquisti (char\* nomefile, Acquisto va[], int\* dim)**

che, ricevuto come parametro di ingresso il nome di un file, provveda ad aprirlo e ne legga il contenuto. Anche in questo caso non è dato sapere a priori il numero di righe presenti nel file di testo, per cui il vettore è allocato nel *main* con dimensione massima MAX\_DIM e passato alla funzione tramite il parametro **va**. La funzione restituisce la dimensione logica tramite il parametro **dim**: come valore di ritorno restituisce 0 in caso di corretta lettura, -1 in caso di errore in apertura del file.

Analogamente al caso precedente, definire una procedura che stampi a video tutte le informazioni relative ad un acquisto:

**void stampaAcquisto (Acquisto a)**



## Esercizio 2

### Esercizio 2 – Cerca peperoncino ([piante.h](#)/[piante.c](#))

Il candidato definisca una funzione:

**int cercaPeperoncino (Peperoncino p[], int dim, char\* codice)**

che preso in ingresso un vettore  $p$  di strutture dati **Peperoncino**, la dimensione  $dim$  del vettore e il *codice* di una varietà di peperoncino, restituisca la posizione nell'array (l'indice) dove si trova il peperoncino ricercato.

Se il codice della varietà cercata non è presente nel vettore, la funzione deve restituire il valore -1.



# Esercizio 3

## Esercizio 3 – Totale scontrino (piante.h/piante.c)

Il candidato definisca una funzione:

**float totaleScontrino(Acquisto a[], int dimA, Peperoncino p[], int dimP, char \* cf)**

che preso in ingresso un vettore  $a$  di acquisti e la sua dimensione, un vettore  $p$  di varietà e la sua dimensione, il codice fiscale  $cf$  di un cliente, completi i seguenti passaggi:

- 1) calcoli l'ammontare in euro della spesa complessiva del cliente;
- 2) aggiorni la struttura dati  $p$  sottraendo il numero di piante acquistate alla corrispondente varietà.
- 3) restituisca come valore di ritorno la spesa complessiva del cliente, applicando uno sconto del 5% se ha acquistato almeno una pianta con un grado di piccantezza maggiore di 100'000 nella scala Scoville.





# Esercizio 4

## Esercizio 4 – Registra acquisto (piante.h/piante.c)

Si realizzi una procedura:

**int registraAcquisto (char\* nomefile, Acquisto \*a)**

che preso in ingresso un singolo acquisto (passato per riferimento) aggiunga una riga in fondo al file di nome nomefile (senza cancellare il contenuto precedente). La funzione deve restituire 0 in caso di corretta scrittura, -1 in caso di errore in apertura del file.



# Esercizio 5

## *Esercizio 5 – Main (main.c)*

Il candidato realizzi un programma in grado di:

- 1) Leggere i prodotti e gli acquisti dai rispettivi file, creando due vettori.
- 2) Stampare la quantità di piante disponibili per la varietà con codice "HABA42" e per la varietà con codice "CIN17".
- 3) Calcolare e mostrare a video il totale dello scontrino dei clienti con codice fiscale "RSSDVD82C08A944D" e "TGLBPP82T19A944E", e stampare la lista aggiornata dei prodotti.
- 4) Leggere da tastiera i dati di un nuovo acquisto e aggiungerli in coda al file degli acquisti.

