



POLITECNICO
MILANO 1863

INFORMATICA A – Prof. Fuggetta

Laboratorio 3

Ing. Paolo Perego, Ph.D.

Ing. Anna Maria Nestorov, MSc.

Esercizio 1

Si scriva un programma C che definisca una struttura che permetta di contenere una serie di dati (struct squadra) del tipo: - nome squadra (stringa di lunghezza 20);

- codice squadra (intero);
- goal fatti (intero);
- goal subiti (intero).

Li si memorizzi in un vettore di strutture "squadre";

- Stampi a terminale tutti i nomi e codici delle squadre che hanno fatto un numero di goal maggiore del numero dei goal subiti.
- Letto a terminale un codice di una squadra stampi a video il nome della squadra, i goal fatti e i goal subiti.

Esercizio 2

Partendo da una struttura dati che descrive un file (NomeFile, data di creazione - 31092017, ultima modifica, dimensione); scrivere un programma che permetta di gestire attraverso un array, tutti i file presenti su disco (Massimo 100 ma possono essere anche meno).

Il programma presenta un menù che permette di:

1. Visualizzare i file che siano stati modificati a partire dalla data inserita dall'utente;
2. Visualizzare lo spazio occupato da tutti i file (attraverso una funzione).
3. Aggiungere un file in coda

Esercizio 3

Scrivere un programma che prenda in ingresso massimo 10 parole (il numero di parole da inserire viene richiesto all'utente all'inizio del programma).

Il programma deve stampare a video le parole inserite dall'utente in ordine alfabetico (per semplicità considerare solo la prima lettera della parola).

N.B.: Ogni parola-stringa è un array. Occorre quindi utilizzare un array di stringhe. Definire la lunghezza massima della parola = 10

Esercizio 4

Scrivere un programma per la gestione della valutazione di un esame universitario. Il programma deve tenere in considerazione:

- Cognome
- Nome
- Matricola
- Esami [4]
- Numero esami sostenuti
- Promosso (0 non promosso - 1 promosso)

Ogni Esame contiene la data dell'esame e la valutazione. Il programma tiene a memoria al massimo 4 prove (array di 4). Inizializzare l'array di esami.voto a -1 (non sostenuti).

Il programma deve avere un menù (usare switch-case) che permetta di:

Inserire un nuovo studente (senza inserire alcun esame). In automatico la casella promosso è definita 0.

Aggiungere un esame cercando lo studente per matricola

Stampare cognome, matricola e voto degli studenti che hanno sostenuto un esame (viene richiesto di inserire un numero da 1 a 4 per selezionare l'esame)

Stampare cognome, matricola e voto degli studenti promossi.

Il campo Promosso deve essere automaticamente modificando quando inserito un voto ≥ 18

Esercizio 5

Scrivere un programma che permetta di inserire un testo (compresi gli spazi) di al massimo 1024 caratteri. Attraverso un menu, il programma deve:

- Inserire un nuovo testo
- Cercare una parola inserita dall'utente e invertirne le lettere (la parola può essere presente più di una volta)
- Cercare una parola e renderla tutta maiuscola (la parola può essere presente più di una volta)
- Stampare il testo

Effettuare la ricerca della parola scrivendo una funzione.

Esercizio 6

Scrivere un programma che, definita una matrice di 2 dimensioni, sia in grado di memorizzare gli appuntamenti di una settimana.

```
Agenda[giorni][appuntamento][100]
```

Definire i giorni della settimana attraverso enum. Gli appuntamenti sono stringhe di lunghezza massima 100 e non contengono l'orario.

L'utente deve:

- Poter inserire un nuovo appuntamento scegliendo il giorno
- Stampare tutti gli appuntamenti divisi per girono
- Stampare gli appuntamenti di un singolo giorno

Per stampare gli appuntamenti di un singolo giorno creare una funzione in modo che possa essere usata ai punti B e C.