



**FONDAMENTI DI INFORMATICA – APPELLO DEL 6 SETTEMBRE 2018 - COMPITO 1**

Cognome e Nome	Matricola	Aula	Postazione
----------------	-----------	------	------------

**SEZIONE 1: PROVA AL CALCOLATORE (VALORE MAX 23 PUNTI)**

Implementare una applicazione console in linguaggio ANSI C per la gestione delle iscrizioni in una palestra. L'applicazione ha il compito di gestire le iscrizioni, le organizzazione dei corsi e i relativi incassi.

Ogni iscrizione è univocamente identificata da un codiceIscrizione. Il programma carica inizialmente l'archivio dei dati da un file di testo (un esempio è fornito nel seguito) in una lista semplice (`lista_iscrizione`) e successivamente copia il contenuto in una lista (`lista_attivita`) in cui ogni nodo contiene il nome dell'attività e gli iscritti di quella attività.

Il codiceTessera individua unicamente una persona. Per la stessa persona possono essere presenti più iscrizioni a diverse attività con codiceIscrizione diverso.

Specifica della struttura dati:

Iscrizione, (elemento della `lista_iscrizione`):

- `codiceIscrizione` (stringa di 12 caratteri utili)
- `eta` (intero)
- `Cognome` (stringa di 20 caratteri utili senza spazi)
- `codiceTessera` (stringa di 6 caratteri)
- `attivita` (stringa di 12 caratteri utili)

Elemento della `lista_attivita`:

- `attivita` (stringa di 12 caratteri utili)
- `elenco degli iscritti per tale attivita` (lista degli iscritti a tale attività)
- `numero degli iscritti con meno di 18 anni`

Elenco delle operazioni/funzioni che devono essere implementate:

(Lo studente noti che il non rispetto delle specifiche nei quesiti 1 e 2, quindi la scelta di un nome della funzione diverso, di parametri differenti o di valori di ritorno differenti sono considerati errori gravi a prescindere del fatto che la funzione svolga quanto richiesto).

1. Funzione di caricamento da file di testo dei dati contenuti nel file. La funzione da implementare deve rispettare la seguente specifica:

nome della funzione: `carica_iscrizioni`;  
valore restituito: `numero di elementi caricati`  
parametri: `nome del file, lista delle iscrizioni`

Lo studente scelga sia il tipo dei dati che i meccanismi di passaggio dei parametri più opportuni per rispettare la specifica.

2. Funzione di modifica dell'attività (devono essere aggiornate tutte le istanze della attività, cioè tutte quelle con lo stesso nome attività). Specifica della funzione:

nome della funzione: `modifica_iscrizioni`  
valore restituito: `0 se non sono state effettuate modifiche 1 se è stata effettuata almeno una modifica`;  
parametri: `lista_info, attivitaVecchia, nuovaAttività, numero di istanze modificate`.  
(NB: il `numero di istanze modificate` è un parametro di uscita che la funzione deve produrre e non visualizzare).

Lo studente scelga sia il tipo dei dati che i meccanismi di passaggio dei parametri più opportuni per rispettare la specifica.

3. Funzione di creazione della lista di liste: Crea la lista delle iscrizioni separate per attività (`lista_attivita`) a partire dalla `lista_iscrizione`. Ogni elemento della lista delle attività contiene il nome dell'attività, il numero degli iscritti con meno di 18 anni e la lista completa degli iscritti a tale attività.
4. Ricerca della attività con più iscritti Funzione che, operando sulla lista delle attività restituisce il nome dell'attività con più iscritti e quella con più iscritti con meno di 18 anni. Nell'ipotesi di più attività con lo stesso valore di iscritti restituisce la prima trovata;
5. Funzione che visualizza il contenuto dell'intera `lista_attivita`
6. Programma principale dotato di un menù testuale che permetta all'utente di usare tutte le funzioni implementate ed eventualmente di inserire i parametri necessari ad eseguire le operazioni richieste.



UNIVERSITÀ  
degli STUDI  
di CATANIA

DIPARTIMENTO di INGEGNERIA  
ELETTRICA ELETTRONICA  
e INFORMATICA

CORSO DI LAUREA in  
INGEGNERIA ELETTRONICA  
INGEGNERIA INFORMATICA

### Struttura del file Iscrizioni

Le informazioni nel file per ciascun dato si trovano distribuite su due righe

<i>codiceIscrizione</i>	<i>attività</i>	
<i>codiceTessera</i>	<i>età</i>	<i>Cognome</i>

Esempio di file:

I123	pallavolo	
T1	12	Amato
I234	basket	
T3	20	Pappalardo
I345	calcio	
T34	23	Papale
T789	calcio	
T1	12	Amato
I989	pallavolo	
T2	21	Dante
I231	hockey	
T56	13	Manzoni



**SEZIONE 2: DOMANDE TEORICHE (VALORE MAX 5 PUNTI)**

**DOMANDA 1 (PUNTI 0,5)**

Cosa è il file system?

**DOMANDA 2 (PUNTI 0,5)**

Dato il seguente frammento di codice, cosa viene visualizzato sullo schermo?

```
int main(void) {  
    int i = -10, k;  
    while (i < 0) {  
        k = -i;  
        for (; k > 0; --k)  
            printf("(%d)(%d)", i, k--);  
        printf("\n");  
        i = i + 1;  
    }  
}
```

**DOMANDA 3 (PUNTI 0,5)**

Verificare se il programma è corretto e in caso affermativo indicare cosa viene visualizzato sullo schermo. Se invece il programma contiene degli errori, descriverli, proporre una soluzione ed indicare cosa verrebbe visualizzato sullo schermo.

```
#include <stdio.h>  
#include <string.h>  
  
int main(void) {  
    char s1[5] = "pippo";  
    char s2[10] = "pluto";  
    char *s3;  
    s3 = strcat(s1,s2)  
    printf("%s\n", s3);  
}
```



**DOMANDA 4 (PUNTI 1,5)**

Si supponga di avere un vettore di interi contenete i seguenti valori: 200, 127, 123, 89, 45, 34, 23, 15, 8, 1

E' possibile usare l'algoritmo di ricerca binaria in un vettore non ordinato? Quante iterazioni vengono effettuate dall' algoritmo di ricerca binaria per verificare che il valore 60 non è presente? Indicare il valore trovato ad ogni iterazione.

**DOMANDA 5 (PUNTI 2)**

Si supponga di costruire un albero binario di ricerca secondo la seguente sequenza di inserimento:

19, 10, 18, 50, 0, 124, 31, 80

Disegnare l'albero che si ottiene con l'inserimento.

Schematizzare l'algoritmo di visita in ordine simmetrico e scrivere la sequenza di valori ottenuti quando si effettua la visita



**FONDAMENTI DI INFORMATICA – APPELLO DEL 6 SETTEMBRE 2018 - COMPITO 2**

Cognome e Nome	Matricola	Aula	Postazione
----------------	-----------	------	------------

**SEZIONE 1: PROVA AL CALCOLATORE (VALORE MAX 23 PUNTI)**

Implementare una applicazione console in linguaggio ANSI C per la gestione dei dati meteorologici acquisiti da una serie di stazioni localizzate in diverse località. Ogni stazione fornisce i dati riassuntivi relativi di una intera giornata relativamente alla temperatura media ed al livello di precipitazioni. Tali informazioni sono memorizzate in un file di testo (un esempio di file è riportato alla fine del testo) e vengono caricate in una lista di misurazioni. Successivamente, a partire da tale lista, viene costruita la lista delle località in cui ogni nodo colleziona tutte le misurazioni relative a quella specifica località. Un nodo della lista delle località contiene il nome della località, il numero di giorni monitorati e l'elenco delle misure effettuate giornalmente in quella località.

Specifiche della struttura dati

Misurazione

- Località (stringa di 15 caratteri utili senza spazi)
- Data (stringa nel formato aaaammgg)
- Temperatura media (valore reale)
- Millimetri di pioggia (valore intero)

Meteo Località

- Località (stringa di 15 caratteri utili senza spazi)
- Numero di misurazioni (valore intero)
- Elenco delle misurazioni (lista di Misurazione)

Elenco delle operazioni/funzioni che devono essere implementate

(Lo studente noti che il non rispetto delle specifiche nei quesiti 1 e 2, quindi la scelta di un nome della funzione diverso, di parametri differenti o di valori di ritorno differenti sono considerati errori gravi a prescindere del fatto che la funzione svolga quanto richiesto).

1. Funzione di caricamento da file di testo delle misurazioni nella lista delle misurazioni. La funzione da implementare deve avere la seguente struttura:  
nome della funzione: *CaricaMisurazioni*  
valore restituito: *numero di misurazioni caricate*  
parametri: *nome del file, lista delle misurazioni*
2. Funzione di ricerca della temperatura più elevata. La funzione da implementare deve avere la seguente struttura:  
nome della funzione: *TemperaturaPiùElevata*  
valore restituito: *0 se la lista delle misurazioni è vuota, 1 altrimenti*  
parametri: *lista delle misurazioni, misurazione*  
(NB: *misurazione* è un parametro di uscita - contenente tutti i dati della misurazione - che la funzione deve produrre e non visualizzare, in caso di più misurazioni con la stessa temperatura massima, restituire solo una di esse).
3. CreazioneListaMeteoLocalita: Crea la lista del meteo per località a partire dalla lista delle misurazioni. Ogni elemento della lista del meteo per località contiene il nome della località, il numero complessivo di misurazioni e l'elenco misurazioni relative a quella specifica località.
4. VisualizzaTutto: Visualizza, per ogni località, l'elenco delle misurazioni relative a quella località.
5. LocalitàPiuPiovosa: Dato un periodo di osservazione definito da una data di inizio e da una data di fine (parametri di ingresso della funzione), restituisce (come parametri di uscita) il nome della località con il livello di precipitazione medio maggiore ed il valore di tale livello nel periodo di osservazione dato.
6. Programma principale dotato di un menù testuale che permetta all'utente di usare tutte le funzioni implementate ed eventualmente di inserire i parametri necessari ed eseguire le operazioni richieste.

Esempio dei dati nel file di ingresso

Catania 20180904 25 0  
Palermo 20180904 27 0  
Catania 20180905 22 2  
Palermo 20180905 25 0



**SEZIONE 2: DOMANDE TEORICHE (VALORE MAX 5 PUNTI)**

**DOMANDA 1 (PUNTI 0,5)**

Descrivere il ruolo dei registri in una CPU

**DOMANDA 2 (PUNTI 0,5)**

Dato il seguente frammento di codice, cosa viene visualizzato sullo schermo?

<pre>int main(void) {     int i = -2, k;     while (i &lt;= 0) {         k = -i;         for (; k &gt; 0; --k)             printf("(%d)(%d)", i, k);         printf("\n");         i = i + 1;     } }</pre>	
---	--

**DOMANDA 3 (PUNTI 0,5)**

Verificare se il programma è corretto e in caso affermativo indicare cosa viene visualizzato sullo schermo. Se invece il programma contiene degli errori, descriverli, proporre una soluzione ed indicare cosa verrebbe visualizzato sullo schermo.

<pre>#include &lt;stdio.h&gt; #include &lt;string.h&gt;  int main(void) {     char s1[] = "topolino";     char s2[5];     strcpy(s2, s1);     printf("%s %d\n", s2, strlen(s2)); }</pre>	
--	--



**DOMANDA 4 (PUNTI 1,5)**

Si supponga di avere un vettore di interi contenete i seguenti valori: 2, 4, 8, 25, 33, 57, 68, 80, 93, 99

E' possibile usare l'algoritmo di ricerca binaria in un vettore non ordinato? Quante iterazioni vengono effettuate dall' algoritmo di ricerca binaria per verificare che il valore 103 non è presente? Indicare il valore trovato ad ogni iterazione.

**DOMANDA 5 (PUNTI 2)**

Si supponga di costruire un albero binario di ricerca secondo la seguente sequenza di inserimento:

18, 2, 24, 10, 0, 48, 62, 85

Schematizzare l'algoritmo di visita in ordine simmetrico e scrivere la sequenza di valori ottenuti quando si effettua la visita?