
PROGRAMMAZIONE I

A.A. 2019 - 2020
CORSO DI LAUREA IN INFORMATICA
DOCENTI: G. DE MARCO R. ZIZZA

PROVA UNICA A DISTANZA - 6 LUGLIO 2020

Consegna

Fase 1 (carta e penna). Creare un file pdf (denominato `cognome_nome.pdf`) contenente la scansione di tutti i fogli su cui sono stati svolti gli esercizi e inviarlo come allegato all'indirizzo di consegna.

Fase 2 (implementazione). L'implementazione dell'esercizio di programmazione svolto durante la prima fase va svolta in un file C (denominato `cognome_nome.c`) che dovrà essere inviato all'indirizzo di consegna entro le ore 19:00 del 6 luglio 2020.

Indirizzo di consegna

`gianluca.demarco@gmail.com`

Indicare "esame programmazione" come oggetto e inserire nome, cognome e matricola nel testo del messaggio.

Teoria

Esercizio 1 *Il seguente frammento di codice è corretto? Se lo è, indicare cosa fa, altrimenti dire qual è l'errore.*

```
char *s = "ciao mondo";  
s = s+5;  
*s = 'M';
```

. ~ .

Esercizio 2 *Cosa stampa il seguente programma? Spiegare il compito svolto dalla funzione f.*

```
#include <stdio.h>
```

```
int f(int n){  
    if (n < 2)  
        return 0;  
    int x = 2;  
    while (x*x <= n){  
        if (n%x == 0)  
            return 0;  
        x++;  
    }  
    return 1;  
}
```

```
int main () {  
  
    printf("%d %d\n", f(33), f(13));  
    return 0;  
}
```

. ~ .

Esercizio 3 Cosa stampa il seguente programma? Spiegare il compito svolto dalla funzione *f*.

```
#include <stdio.h>

void f(char *s, char *t) {
    char ch;
    while (*s && *t && *s == *t) {
        s++;
        t++;
    }
    while (*t && *s) {
        ch = *s;
        *s = *t;
        *t = ch;
        s++;
        t++;
    }
}

int main () {
    char s[] = "buongiorno", t[] = "buonanotte";
    printf("%s %s\n", s, t);
    f(s, t);
    printf("%s %s\n", s, t);
    return 0;
}
```

. ~ .

Programmazione

Esercizio 4 Scrivere un programma *laghi* che permetta di esaminare la lista dei laghi italiani memorizzata in un file *laghi.txt*. Il programma può essere usato in tre modi.

1) Prende in input da linea di comando il nome di un lago. Stampa: il nome del lago, la sua area in km^2 , la sua profondità massima, la sua altitudine e, infine, la lista delle regioni e/o stati in cui si trova. Esempio:

```
> ./laghi Bolsena
Lago di Bolsena 114 146 305 Lazio
```

2) Prende in input da linea di comando un intero indicante un'area in km^2 . Stampa: l'elenco dei laghi la cui area è almeno quella indicata; ciascun lago sarà seguito dalla sua area in km^2 . Esempio:

```
> ./laghi 100
Garda 370
Maggiore 212
Como 146
Trasimeno 128
Bolsena 114
```

3) Prende in input da linea di comando il carattere *d* seguito da un intero indicante un'area in km^2 . Stampa: tra tutti i laghi aventi un'area non inferiore a quella indicata, quello che ha la maggiore profondità massima; il nome del lago sarà seguito dalla sua area e dalla sua profondità massima. Esempio:

```
> ./laghi d 100
Como 146 410
```

Il file *laghi.txt* contiene l'elenco dei laghi italiani ed ha il seguente formato. Ogni riga corrisponde ad un lago e presenta nell'ordine: il suo nome seguito da una virgola, la sua area in km^2 , la sua profondità massima in m, la sua altitudine in m.s.l.m e infine, le regioni e/o stati in cui si trova. Si noti che il numero di spazi fra gli elementi è indefinito.

```
Garda, 370 346 65 Lombardia, Veneto, Trentino-Alto Adige
Maggiore, 212 372 193 Lombardia, Piemonte, Svizzera
Como, 146 410 197 Lombardia
Trasimeno, 128 6 258 Umbria
Bolsena, 114 146 305 Lazio
Iseo, 65 251 185 Lombardia
Varano, 60 6 0 Puglia
...
```

Informazioni utili per l'implementazione

Attenzione: Per la stampa dell'output, ci si attenga strettamente alle indicazioni date sopra e illustrate negli esempi, evitando di introdurre frasi non richieste, come: "I laghi ... sono ..., la profondità ... è ..., ecc. **Soluzioni che ignorino tali indicazioni, rischiano di essere penalizzate o di non essere prese in considerazione!**

Si ricordi che lo specificatore di formato `% [^ set]` consente di far corrispondere ogni sequenza di caratteri non presente in *set*.

. ~ .