

Esercizio 1

Scrivere un programma che riceve in ingresso una stringa di al massimo 20 caratteri. Successivamente, occorre sostituire nella stringa per ogni carattere nell'intervallo 'a'...'z' il corrispondente carattere maiuscolo ('A'...'Z'). Qualsiasi altro carattere viene sostituito da uno spazio. **Obbligatorio:** creare e utilizzare un sottoprogramma che riceve in ingresso, mediante un parametro, un carattere e restituisce, mediante un altro parametro, il corrispondente maiuscolo. Il valore di ritorno della funzione è 1 se è stato restituito un carattere maiuscolo, 0 in tutti gli altri casi (con il valore del secondo parametro non significativo). Infine il programma stampa la stringa modificata.

NOTA: ogni carattere corrisponde ad un codice ASCII, due caratteri consecutivi corrispondono a due valori ASCII consecutivi. Quindi, il risultato di `car-'a'` è la distanza in caratteri del carattere `car` da 'a'. Esempio: `'d'-'a'=3`

Esempio di esecuzione:

aBc1d3e,f -> "A C D E F"

Esercizio 2

Scrivere un programma che acquisisca le votazioni espresse da 7 giudici, ognuna mediante un numero da 0 a 10, e calcoli il punteggio finale come media di tutte le votazioni espresse, escluse quella più alta e quella più bassa. Qualora le votazioni massime fossero in numero superiore ad una, solo una tra esse verrà esclusa dal calcolo della media. Analoga considerazione va fatta per le votazioni minime. **Obbligatorio:** creare e utilizzare due sottoprogrammi per calcolare rispettivamente la posizione del giudice che ha espresso il voto massimo e quello minimo. Ogni sottoprogramma riceve in ingresso, mediante un parametro, un array di interi, e restituisce come valore di ritorno rispettivamente la posizione nell'array col valore massimo e minimo.

Esempio di esecuzione:

Ingresso: 0 2 1 7 8 8 3

Uscita: 4.2

Esercizio 3

Creare un programma in grado di gestire un dizionario Italiano – Inglese, presente su un file di testo. Il programma dovrà mostrare un menù a video con le seguenti funzionalità (**obbligatorio** implementare le funzionalità come sottoprogrammi):

1. cerca un vocabolo nel dizionario
2. inserisci un nuovo vocabolo

Il primo sottoprogramma riceve in ingresso, mediante un parametro, un vocabolo italiano di cui si vuole ottenere la traduzione in inglese, legge il file dizionario e nel caso in cui venga trovata la parola, riporta a video la traduzione. Il secondo sottoprogramma invece riceve in ingresso, mediante due parametri, un nuovo vocabolo italiano e la relativa traduzione in inglese, e li salva nel file dizionario (mantenendo le informazioni già presenti). Un singolo vocabolo non potrà essere più lungo di 20 caratteri.

Esempio per il file dizionario.txt

mela = apple
rosso = red
casa = house

Esercizio 4

Si sviluppi un programma per memorizzare gli orari delle lezioni gestendo i dati:

- nome corso: max 20 caratteri
- giorno della settimana: 1 cifra da 0 a 6
- orario: ora inizio ed ora fine

Un corso potrà avere più lezioni nell'arco della settimana, però non sarà possibile avere più di 4 turni settimanali per un corso per un massimo di 5 corsi.

Lo studente dovrà quindi definire un'opportuna struttura dati per la gestione delle informazioni suddette e quindi implementare il programma che proponga un menù per:

- Inserire le informazioni per i corsi;
- Stampare gli orari settimanali dato il nome del corso

Obbligatorio, implementare 3 sottoprogrammi. Il primo sottoprogramma riceve in ingresso, mediante un parametro, un array di opportuna struttura dati (e qualsiasi altro parametro è ritenuto necessario), chiede all'utente le informazioni per il nuovo corso da inserire, e restituisce la posizione dell'inserimento nell'array, -1 se l'operazione fallisce. Il secondo sottoprogramma riceve in ingresso, mediante parametri, un array di opportuna struttura dati (e qualsiasi altro parametro è ritenuto necessario), il nome del corso da cercare, e restituisce la posizione nell'array dove si trova il corso, altrimenti, se il corso non esiste, restituisce -1. Il terzo sottoprogramma riceve in ingresso, mediante parametri, un array di opportuna struttura dati (e qualsiasi altro parametro è ritenuto necessario), e un intero pos, stampa tutte le informazioni della posizione pos nell'array.

Esempio di esecuzione:

```
1. inserisci corso
2. stampa orario
3. esci
-->1
inserisci nome per il nuovo corso :infoA
inserisci il numero dei turni :1
inserisci il giorno della settimana (dom=0, .., sab=6) 4
inserisci l'ora d'inizio 8
inserisci l'ora di fine 12
```

```
1. inserisci corso
2. stampa orario
3. esci
-->2
Inserisci nome corso :infoA
Trovato corso: infoA
Giorno della settimana 4
Orario 8 - 12
```