

Fondamenti di Informatica ♦ 2021-22

Esercitazione - Analisi delle soluzioni #3

24 novembre 2021

Esercizio 1 – tema d'esame 4 settembre 2018

Scrivere un programma in C ANSI che chiede all'utente il nome di tre file di testo ASCII (ognuno di al più 80 caratteri, comprensivi di percorso ed estensione) ed un intero: i primi due file sono i file sorgente, il terzo è il file destinazione. Il programma crea il terzo file concatenando il contenuto dei due file in ingresso: se l'intero fornito come terzo parametro vale 0, il programma concatena il contenuto del secondo file al primo, se l'intero fornito vale 1, il programma concatena il contenuto del primo al secondo. Nel caso in cui l'intero valga un valore diverso da 0 e 1, il programma non fa nulla.

Esercizio 2 – tema d'esame 3 luglio 2017

Scrivere un programma in C ANSI che acquisisca una stringa `str` (priva di spazi e lunga al più 500 caratteri). La stringa `str` contiene una sequenza ordinata (ordine alfabetico) di caratteri non necessariamente tutti distinti (può cioè contenere delle ripetizioni). Per ogni carattere `c` contenuto nella stringa `str`, a partire dall'ultimo fino ad arrivare al primo, il programma visualizza (senza lasciare spazi) il carattere `c`, seguito dal numero di volte che questo compare in `str`.

Per esempio, data la stringa `aabchhhhhpppzzzzz` il programma visualizza `z5p3h5c1b1a2`.

Esercizio 3 – tema d'esame 17 febbraio 2020

Scrivere un sottoprogramma `conta` che riceve in ingresso due stringhe, `info` e `cars`. Il sottoprogramma `conta` e restituisce al chiamante il numero di volte che ogni carattere presente in `cars` compare nella stringa `info`. Si crei una opportuna struttura dati per restituire al chiamante il risultato dell'elaborazione. Nella stringa `cars` non ci sono caratteri ripetuti. Per esempio, se le stringhe sono le seguenti:

`info: TTGCAACAGCCTTAACAGCTTTGAATGACATGGGAAAGGTAAGAAA`

`cars: ACGT`

il sottoprogramma restituisce una struttura dati che contiene i valori `18 7 11 10`.

Non sovradimensionare la struttura dati, deve essere definita su misura per contenere il numero di contatori strettamente necessari.

Esercizio 4 – tema d'esame 5 febbraio 2021

Scrivere un sottoprogramma `distanzaLineare` che riceve in ingresso le seguenti informazioni:

- ♦ `start`: il primo valore (valore reale)
- ♦ `stop`: l'ultimo valore (valore reale)
- ♦ `numero`: il numero di valori da distribuire nell'intervallo `[start, stop]` (valore intero)
- ♦ `escludistop`: 1 se si vuole escludere il valore `stop`, ossia se i valori devono coprire l'intervallo `[start, stop)`, 0 altrimenti

Il sottoprogramma crea e restituisce un insieme di numeri uniformemente distribuiti nell'intervallo `[start, stop]` (oppure `[start, stop)`).

Se il sottoprogramma riceve in ingresso i valori: `2.0, 3.0, 5, 0` l'insieme dei dati conterrà i seguenti valori:

`2.0 2.25 2.50 2.75 3.0`

Se il sottoprogramma riceve in ingresso i valori: `2.0, 3.0, 5, 1` l'insieme dei dati conterrà i seguenti valori:

`2.0 2.2 2.4 2.6 2.8`

Esercizio 5 – tema d'esame 13 settembre 2017

I numeri di Catalan sono una successione C_n di interi positivi. Il nome di questi numeri è stato scelto in onore del matematico belga Eugène Charles Catalan (1814-1884) che li aveva studiati intorno al 1838 per stabilire in quanti modi si può ricondurre il prodotto di n fattori ad una successione di prodotti a coppie. I numeri di Catalan si possono ottenere ricorsivamente, secondo la seguente relazione:

$$C_0 = 1$$

$$C_n = \sum_{i=0}^{n-1} C_i \times C_{n-i-1}$$

Si scriva un sottoprogramma ricorsivo `cat` che, ricevuto un numero n come parametro formale, ritorna il valore di C_n .

Si scriva poi il programma principale che acquisisca dall'utente un numero intero positivo x (e finché non è tale lo richiede) e quindi calcola e visualizza i primi $x+1$ numeri di Catalan, da C_0 a C_x utilizzando il sottoprogramma `cat`.