

Fondamenti di Informatica ◇ 2021-22

Laboratorio #2 del 19-10-2021

Somma a k

Scrivere un programma che acquisisce una sequenza di al più 100 valori interi e un intero strettamente positivo k . L'acquisizione della sequenza termina quando l'utente inserisce un numero negativo o nullo (che verrà comunque fornito, al più come 101esimo valore, prima di k). Il programma visualizza 1 se la sequenza contiene due valori tali che la loro somma sia k , 0 altrimenti, seguito dal carattere 'a-capo'.

Ingresso/Uscita:

input: una sequenza di al più 101 valori interi

output: un intero (seguito da un carattere 'a-capo')

Alcuni casi di test per il collaudo:

input: 10 15 3 7 -4 17

output: 1

input: 2 5 6 55 10 -11 100

output: 0

input: 2 2 3 -8 7

output: 0

Solo in ordine

Scrivere un programma che acquisisce una sequenza di al più 20 valori interi, chiedendo all'utente inizialmente quanti valori vorrà fornire, num . Il programma acquisisce num valori e memorizza in una opportuna struttura dati la sequenza di valori i cui elementi sono strettamente crescenti, trascurando i valori che risultano non essere ordinati. Nella struttura dati, quindi ci sono solo dati ordinati. Al termine dell'acquisizione il programma visualizza la lunghezza della sequenza, seguita, su una nuova riga, dalla sequenza stessa. L'utente inserirà sempre un numero di valori coerente con la richiesta (ossia la quantità indicata da num è sempre rispettata e num non è maggiore di 20).

Ingresso/Uscita:

input: sequenza di interi

output: sequenza di interi

Alcuni casi di test per il collaudo:

input: 10

3 1 4 5 -1 3 1 4 5 -1

output: 3

3 4 5

input: 6

-1 3 -1 3 -1 3

output: 2

-1 3

input: 8

9 8 7 6 5 4 3 2

output: 1

9

Mix di due array ordinati

Scrivere un sottoprogramma che acquisisce due sequenze di valori interi, ciascuna di 20 elementi. Il programma ordina le due sequenze in senso crescente, quindi visualizza la sequenza dei valori acquisiti, in senso crescente e senza ripetizioni. Al termine dell'esecuzione, le due sequenze sono ordinate.

Ingresso/Uscita:

input: quaranta sequenze di numeri positivi

output: una sequenza di numeri positivi

Alcuni casi di test per il collaudo:

input: -1 2 4 2 5 6 8 1 0 7 3 4 9 -9 9 9 9 9 9 9
 7 3 4 5 6 7 8 9 2 1 0 5 6 8 1 0 1 2 4 2

output: -9 -1 0 1 2 3 4 5 6 7 8 9

input: 9 8 7 6 5 4 3 2 1 0 0 0 0 0 0 0 0 0 0 0

input: 1 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2

output: 0 1 2 3 4 5 6 7 8 9

Sottostringa più lunga senza ripetizioni

Scrivere un programma che acquisita una stringa di al più 30 caratteri, individui la sottostringa più lunga in essa contenuta, senza caratteri ripetuti. Il programma visualizza la lunghezza di tale sottostringa, seguita da un carattere 'a-capo'.

Ingresso/Uscita:

input: una stringa

output: un intero

Alcuni casi di test per il collaudo:

input: abcabcbcb

output: 3

input: alfabeto

output: 7

input: bbbbbb

output: 1

versione alternativa con array di contatori a supporto

Quadro di parole

Scrivere un programma che acquisisce un valore intero strettamente positivo `num`, che rappresenta il numero di parole (ciascuna di al più 25 caratteri) che verranno poi fornite, e che comunque non saranno mai più di 20. Il programma acquisisce le `num` parole e le visualizza, una per riga, all'interno di un rettangolo creato dal carattere `*`.

Per esempio, se l'utente fornisce:

```
5
Hello
world
in
un
rettangolo
```

il programma visualizza:

```
*****
*Hello      *
*world      *
*in         *
*un         *
*rettangolo*
*****
```

Ingresso/Uscita:

input: un intero e una sequenza di stringhe

output: una sequenza di caratteri

Stringhe senza ripetizioni

Scrivere un programma che acquisisce una stringa di al più 30 caratteri (compresi eventuali spazi). Il programma calcola e visualizza una stringa cosiffatta:

1. ogni carattere è presente nella stringa riscritta una sola volta, preservandone l'ordine di apparizione rispetto agli altri caratteri. In altre parole, solo la prima occorrenza del carattere nella stringa viene riportato nella stringa risultato;
2. ogni spazio è preservato.

Il programma visualizza la stringa così manipolata, seguita dal carattere a capo.

Ingresso/Uscita:

input: una stringa di al più 30 caratteri

output: una stringa seguita dal carattere 'a-capo'

Alcuni casi di test per il collaudo:

input: aaabbbccddd effffg

output: abcd efg

input: abracadabra

output: abrcd
