

Fondamenti di Informatica ◇ 2021-22

Laboratorio #3 del 26-10-2021

Vocaboli con doppie

Scrivere un sottoprogramma `conta` che ricevuta in ingresso una stringa `conta` e restituisce il numero di vocaboli contenenti doppie. La stringa ricevuta in ingresso contiene una frase scritta in italiano con vocaboli separati esclusivamente da spazi.

Ingresso/Uscita:

input: una stringa

output: un intero

Alcuni casi di test per il collaudo:

input: Prova di collaudo del programma che ho appena fatto per vedere se funziona
output: 4

input: accipicchia quasi non mi accorgo di questa casistica per verificare che tutto funzioni
output: 3

input: effettivamente
output: 1

input: questa frase e un poco piu' lunga delle altre per poter avere piu' vocaboli con doppie e assicurarsi che non ci siano particolari casistiche che mandano il programma in errore durante la sua esecuzione
output: 5

Compleanni

Scrivere un programma che chiede all'utente le date di nascita di 100 persone, rappresentate come tre interi (il giorno, il mese e l'anno). Il programma poi chiede all'utente una data e conta e visualizza il numero di persone nate in quel giorno, e il numero di persone che festeggiano insieme il compleanno in tale data. Il programma visualizza i due valori interi, separati da uno spazio e seguiti da un carattere a capo. Definire un tipo opportuno per rappresentare le date.

Realizzare la propria soluzione organizzandola in sottoprogrammi, come ritenuto più opportuno.

Ingresso/Uscita:

input: 300 interi + 3 intero

output: due interi separati da uno spazio e seguiti da un carattere a-capo

Alcuni casi di test per il collaudo:

input: 13 03 1999
20 01 1995
15 05 1997
21 12 1995
20 01 1994
21 09 2000
21 10 2001
21 10 2000
20 01 1995
12 12 1998
...
20 01 1995

output: 2 3

Cifra k

Scrivere un sottoprogramma `cifra` che riceve come parametri due interi `num` e `k`. Se `k` è strettamente positivo, il sottoprogramma calcola e restituisce la `k`-esima cifra del numero `num` a partire da destra. Nel caso in cui `k` non sia strettamente positivo o `k` sia maggiore del numero effettivo di cifre di `num`, il sottoprogramma restituisce `-1`.

Scrivere un programma che chiede all'utente i due valori `num` e `k` ed invoca il sottoprogramma `cifra` visualizzando poi il risultato.

Ingresso/Uscita:

input: due interi

output: un intero

Alcuni casi di test per il collaudo:

input: 1425 2

output: 2

input: -9064663 1

output: 3

input: 9064663 -1

output: -1

input: 9064663 6

output: 0

Numeri perfetti

Scrivere un programma che riceve in ingresso un array di 10 valori numeri interi. Costruire un nuovo array dove in corrispondenza di ogni elemento dell'array iniziale si mette:

- ◇ 0 se l'elemento *i* è non strettamente positivo,
- ◇ 1 se l'elemento *i* è un numero *perfetto*,
- ◇ 2 se l'elemento *i* è *abbondante*,
- ◇ 3 se l'elemento *i* è *difettivo*

Per definizione un numero è *perfetto* se corrisponde alla somma dei suoi divisori, escluso se stesso; *abbondante* se è minore della somma dei suoi divisori, escluso se stesso; altrimenti è *difettivo*.

Ad esempio: 6 e 28 sono *perfetti*, 12 e 18 sono *abbondanti*, mentre 8 e 10 sono *difettivi*.

Il programma visualizza la nuova matrice seguita da a-capo. Inoltre, il programma visualizza il numero di numeri perfetti, difettivi e abbondanti contenuti nella matrice di origine, separati da spazio e seguiti dal carattere a-capo a-capo.

Realizzare la propria soluzione organizzandola in sottoprogrammi, come ritenuto più opportuno.

Ingresso/Uscita:

input: 10 valori interi

output: 10 valori interi seguiti da carattere a-capo 3 valori interi seguiti da carattere a-capo

Alcuni casi di test per il collaudo:

input: 12 28 -1 0 6 10 18 -8 0 5

output: 2 1 0 0 1 3 2 0 0 3

2 2 2

Lettere a scorrere

Scrivere un sottoprogramma che riceve in ingresso una stringa *sequenza* ed un valore intero *chiave* che modifica la stringa ricevuta in ingresso sostituendo ogni carattere con quello che lo segue di *chiave* posti. Se si supera il carattere *z* si riparte dal carattere *a*. Si ipotizzi che i caratteri presenti nella stringa iniziale siano o tutti maiuscoli o tutti minuscoli.

Ingresso/Uscita:

input: una stringa e un intero

output: modifica della stringa di partenza

Alcuni casi di test per il collaudo:

input: "validazione" 4

output: "zepmhedmsri"

input: "ZANZARA" 1

output: "ABOABSB"
