

Laboratorio di Programmazione
Preparazione al II Compitino

11 gennaio 2008

Cognome Nome

Matricola

1. Scrivete un programma che operi come segue: legge un primo numero intero (diciamo n) e poi legge n numeri interi. Potete assumere che $n \leq 100$. Al termine della lettura, deve stampare gli interi inseriti nel seguente ordine: il primo, l'ultimo, il secondo, il penultimo e così via. Ad esempio, se l'utente inserisce *7 61 300 506 61 601 503 20* il programma deve stampare *61 20 300 503 506 601 61*.

2. Scrivete una funzione che, data una stringa e un carattere, restituisca la lunghezza della sottosequenza compresa fra la prima e l'ultima occorrenza del carattere, oppure -1 se il carattere non compare. Ad esempio, se la stringa è "carattere davvero non facile" e il carattere è 'a', la funzione deve restituire 23 (la lunghezza della stringa "arattere davvero non fa").

3. Considerate la seguente struttura:

```
typedef struct {
    int larghezza , lunghezza , altezza ;
    double peso_specifico ;
    char colore [20];
} scatola ;
```

che specifica i dati di una scatola (le misure, in centimetri; il peso, in grammi per centimetro cubo; e il colore, una stringa). Scrivete una funzione che riceva come argomenti:

- un array di scatole;
- un numero n (il numero di elementi dell'array da considerare);
- una stringa rappresentante il nome di un colore.

La funzione deve restituire il peso complessivo delle scatole nell'array.

4. Scrivete un programma che, ricevendo sulla riga di comando due stringhe, determini se la prima ha la seconda come prefisso, stampando “sì” oppure “no”. Ad esempio, se supponete che il programma si chiami `prefisso`, un tipico esempio di esecuzione potrebbe essere

```
[gnagna]>./prefisso cane ca  
sì  
[gnagna]>./prefisso cane co  
no  
[gnagna]>
```

5. Scrivete una funzione che prende come argomento un `double x`, un puntatore a un intero e un puntatore a un `double`, e che metta in questi ultimi rispettivamente la parte intera e la parte frazionaria di x .

Ora scrivete un `main` che legga (mediante una `scanf`) un numero `double`, e ne stampi la parte intera e la parte frazionaria, usando la funzione scritta.