

Laboratorio di algoritmi e strutture dati

Docente: Violetta Lonati

Esercizi introduttivi sulle funzioni in C

1 Potenza

Scrivete una funzione avente due parametri interi b ed e che calcoli la potenza b^e .

2 Primo

Scrivete una funzione con parametro un intero n che stabilisca se n è un numero primo.

3 Garibaldi fu ferito

Scrivete un programma che legga un carattere, uno spazio e quindi una sequenza di caratteri minuscoli terminati da `.` e che stampi quanto ha letto dopo il primo spazio, ma sostituendo tutte le vocali con il primo carattere letto. Per farlo, usate una funzione che, dati due caratteri, restituisca il primo carattere se il secondo è una vocale minuscola, altrimenti restituisca il secondo carattere.

Esempio di funzionamento

```
Stringa: u garibaldi fu ferito, fu ferito in una gamba.  
gurubuldu fu furutu, fu furutu un unu gumbu
```

4 Sequenze di Collatz

Considerate la seguente regola: dato un numero intero positivo n , se n è pari lo si divide per 2, se è dispari lo si moltiplica per 3 e si aggiunge 1 al risultato. Quando n è 1 ci si ferma.

Questa semplice regola permette di costruire delle sequenze: la sequenza che si costruisce a partire dal numero n è detta *sequenza di Collatz di n* . Ad esempio, la sequenza di Collatz di 7 è:

7 22 11 34 17 52 26 13 40 20 10 5 16 8 4 2 1

E' un noto problema aperto stabilire se ogni sequenza di Collatz termina (cioè, se arriva a 1).

Scrivete innanzitutto una funzione che, dato un numero, dia il successivo in una sequenza di Collatz. Quindi, inseritela in un programma che chiede all'utente un numero e mostra la sequenza di Collatz del numero (con tanto di lunghezza).

Esempi di funzionamento

```
Numero: 7
7 22 11 34 17 52 26 13 40 20 10 5 16 8 4 2 1
Lunghezza: 17
```

```
Numero: 9
9 28 14 7 22 11 34 17 52 26 13 40 20 10 5 16 8 4 2 1
Lunghezza: 20
```

5 Quadrato magico

Scrivete una funzione `void` che riceva come argomento un intero n dispari, e stampi un quadrato magico di dimensione $n \times n$.

Un quadrato magico è una disposizione dei numeri $1, 2, \dots, n^2$ tale che in ogni riga, in ogni colonna e nelle due diagonali la somma dei numeri sia la stessa.

Memorizzate il quadrato magico in un array bidimensionale, dichiarato localmente alla funzione. Iniziate mettendo il numero 1 al centro della prima riga. Mettete i rimanenti numeri nell'ordine muovendovi in alto di una riga e a destra di una colonna. Se una cella risulta essere già occupata da un altro numero, allora mettete il numero esattamente sotto al numero che lo precede. Ogni volta che raggiungete un bordo, proseguite ripartendo dal lato opposto del quadrato. Ad esempio, se avete messo un numero nella cella di riga 3 e colonna $n - 1$, allora il successivo andrà messo in riga 2 e colonna 0.

Nota: verificate il funzionamento dell'algoritmo e del programma calcolando la somma dei numeri in ogni colonna, riga e diagonale!

Esempio di funzionamento:

```
Inserisci la dimensione del quadrato: 5
17 24 1 8 15
23 5 7 14 16
4 6 13 20 22
10 12 19 21 3
11 18 25 2 9
```

6 Numeri di Fibonacci

I numeri di Fibonacci furono introdotti da Leonardo di Pisa, detto Fibonacci, per descrivere la crescita di una popolazione di conigli (sebbene in modo idealizzato e non plausibile dal punto di vista biologico). Per calcolare il numero di coppie di conigli nella popolazione all' n -esimo mese, egli fece le seguenti assunzioni: al primo mese, c'è una sola coppia di conigli; le coppie neonate diventano fertili a partire dal secondo mese di vita; ogni mese le coppie fertili generano una nuova coppia ciascuna; i conigli non muoiono mai.

Se traduciamo queste ipotesi in formule, chiamato F_n il numero di coppie di conigli presenti all' n -esimo mese, avremo $F_1 = F_2 = 1$ e, per i mesi successivi, $F_n = F_{n-1} + F_{n-2}$, infatti ad ogni mese avremo le coppie del mese precedente, più le nuove coppie generate dalle coppie nate due mesi prima. Il numero di coppie di conigli cresce come $1, 1, 2, 3, 5, 8, 13, 21, \dots$

Scrivete un programma (chiamato `fibonacci`) che, basandosi su una funzione ricorsiva, calcoli il valore di F_n , dove n è letto come primo passo del programma.

Esempio di funzionamento

4 3

Esempio di funzionamento

7 13

Modificate il programma perchè, oltre a stampare F_n , stampi anche il numero di volte che la funzione ricorsiva viene chiamata per calcolare tale valore.

Scrivete una versione del programma che non usi la ricorsione. Prima di impostarlo, ponetevi questa domanda: di quante variabili ho effettivamente bisogno?