

LABORATORIO DI PROGRAMMAZIONE 1
CORSO DI LAUREA IN MATEMATICA
UNIVERSITÀ DEGLI STUDI DI MILANO
2018–2019
15.V.2019

VINCENZO MARRA

INDICE

Parte 1. Input/Output standard dalla console: esercizi facili	2
Esercizio 1	2
<i>Eco carattere per carattere</i>	2
Tempo: 15 min.	2
Esercizio 2	2
<i>Lettura di dati con scanf</i>	2
Tempo: 20 min.	2
Esercizio 3	2
<i>Eco di stringa con fgets</i>	2
Tempo: 15 min.	2
Esercizio 4	2
<i>La procedura leggi_str di Prog1</i>	2
Tempo: 15 min.	3
Parte 2. Ricorsione: Catalan, Delannoy, Hofstadter	4
Esercizio 5	4
<i>Numeri di Catalan</i>	4
Tempo: 20 min.	4
Esercizio 6	4
<i>Tabulazione dei numeri di Catalan</i>	4
Tempo: 25 min.	5
Esercizio 7	5
<i>Numeri di Delannoy</i>	5
Tempo: 20 min.	6
Esercizio 8	6
<i>Tabulazione dei numeri di Delannoy</i>	6
Tempo: 25 min.	6
Esercizio 9	6
<i>Successioni maschio-femmina di Hofstadter</i>	6
Tempo: 30 min.	6

Parte 1. Input/Output standard dalla console: esercizi facili

ESERCIZIO 1

Eco carattere per carattere.

Tempo: 15 min.

Scrivete un programma che esegua l'eco, carattere per carattere, di quanto digitato dall'utente sulla console. Usate la funzione `getchar` di `stdio.h`. Il programma termina quando riceve in ingresso un segnale EOF, che sulla maggior parte dei sistemi è prodotto dalla pressione contemporanea dei tasti CTRL e D.

ESERCIZIO 2

Lettura di dati con `scanf`.

Tempo: 20 min.

Scrivete un programma che presenti all'utente un menu con le voci:

1. Inserisci `char`.
 2. Inserisci `int`.
 3. Inserisci `float`.
 4. Inserisci `double`.
 5. Esci.
- >

dove > indica all'utente che la macchina è in attesa dell'input dell'utente. L'utente inserisce una scelta, che il programma acquisisce come dato di tipo `char` usando `getchar`. Se la scelta è inesistente, il programma stampa un messaggio d'errore, e visualizza nuovamente il menu. Se la scelta è 5, il programma termina. Se la scelta è 1, 2 o 3, il programma legge un dato del tipo appropriato da console usando la funzione `scanf` di `stdio.h`, esegue l'eco del dato acquisito, e visualizza nuovamente il menu. Cosa succede se l'utente digita stringhe che non sono interpretabili come dati del tipo in questione? Sperimentate per acquisire familiarità col comportamento di `scanf`.

ESERCIZIO 3

Eco di stringa con `fgets`.

Tempo: 15 min.

Scrivete un programma che legga una stringa da console e ne esegue l'eco. Usate la funzione `fgets` di `stdio.h`. Il programma termina quando l'utente inserisce la stringa vuota. Usate `BUFSIZ` come lunghezza massima della stringa. La costante `BUFSIZ` è dichiarata in `stdio.h`.

ESERCIZIO 4

La procedura `leggi_str` di Prog1.

```

1  /** Legge una riga da terminale.
2  *
3  * Legge una riga da terminale (stdin) e la acquisisce come char * nella
4  * variabile letta.
5  * Garantisce che la stringa letta sia terminata da '\0'.
6  * msg: Un messaggio per l'utente.
7  * letta: La stringa letta.
8  * Restituisce 0 se si verifica un errore di I/O o se si rileva EOF, e 1
9  * se la lettura va a buon fine.
10 * /
11
12 int leggi_str(char *msg, char *letta)
13 {
14     char *p;
15
16     printf("%s", msg);
17
18     if (fgets(letta, 1024, stdin) != NULL)
19     {
20         if ((p = strchr(letta, '\n')) != NULL)
21             *p = '\0';
22         return 1;
23     }
24     return 0;
25 }

```

FIGURA 1. Il sorgente della procedura `leggi_str` di `Prog1`.

Tempo: 15 min.

La Figura 1 riporta il codice sorgente della procedura `leggi_str` di `Prog1`. Scrivete un programma che usi questa procedura per leggere una stringa ed eseguirne l'eco. (Copiate il sorgente della procedura nel vostro programma, non usate la libreria.) Analizzate il sorgente di `leggi_str` per capire come funziona la procedura. La funzione della libreria standard

`char *strchr(char *s, int c)`

dichiarata in `string.h`, restituisce un puntatore alla prima occorrenza nella stringa `s` del carattere `c`.¹ Nella ricerca è incluso il carattere di terminazione `'\0'`.

¹Abbiamo già detto che `char` è un sottotipo di `int`. Il tipo di `c` è dichiarato come `int` e non come `char`, cosa usuale nella libreria standard del C, come abbiamo visto a lezione.

Parte 2. Ricorsione: Catalan, Delannoy, Hofstadter

I numeri di Catalan

I *numeri di Catalan* prendono il nome dal matematico belga Eugène Charles Catalan (1814–1894). Essi si denotano con $C(n)$, dove n è un intero ≥ 0 , e contano il numero di cammini in una griglia quadrata di dimensione $n \times n$ che partono dall'angolo sud-ovest e arrivano all'angolo nord-est senza mai oltrepassare la diagonale sud-ovest/nord-est della griglia, impiegando solo passi verso nord ed est. Si veda la Figura 2.

I numeri di Catalan soddisfano la relazione ricorsiva:

$$C(n) = \frac{2(2n-1)C_{n-1}}{n+1}$$

con condizione iniziale

$$C(0) = 1.$$

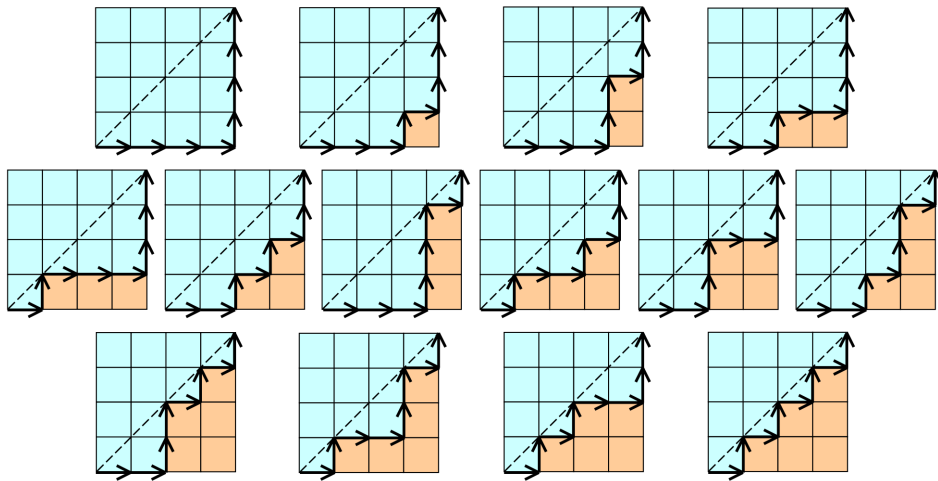


FIGURA 2. I 14 cammini di Catalan sulla griglia 4×4 .

ESERCIZIO 5

Numeri di Catalan.

Tempo: 20 min.

Scrivete un programma ricorsivo che legga in ingresso un intero $n \geq 0$ e calcoli e visualizzi il numero di Catalan $C(n)$. Nel caso in cui l'utente inserisca un valore che non soddisfa la condizione $n \geq 0$, forzate il reinserimento.

ESERCIZIO 6

Tabulazione dei numeri di Catalan.

Tempo: 25 min.

Scrivete un programma che legga in ingresso un intero $n \geq 0$ e calcoli e visualizzi la sequenza $C(0), C(1), \dots, C(n)$ di numeri di Catalan. Per esempio, nel caso in cui $n = 9$ l'output del vostro programma dovrebbe essere:

1, 1, 2, 5, 14, 42, 132, 429, 1430, 4862.

Riutilizzate la funzione ricorsiva scritta per risolvere l'Esercizio 5.

I numeri di Delannoy

I *numeri di Delannoy* prendono il nome dall'ufficiale francese Henri-Auguste Delannoy (1833–1915), matematico dilettante. Essi si denotano con $D(m, n)$, dove m ed n sono interi ≥ 0 , e contano il numero di cammini in una griglia rettangolare di dimensione $m \times n$ che partono dall'angolo sud-ovest e arrivano all'angolo nord-est, impiegando solo passi verso nord, est e nord-ovest. Si veda la Figura 3.

I numeri di Delannoy soddisfano la relazione ricorsiva:

$$D(m, n) = D(m - 1, n) + D(m, n - 1) + D(m - 1, n - 1) \quad (1)$$

con condizioni iniziali

$$D(0, n) = D(m, 0) = 1. \quad (2)$$

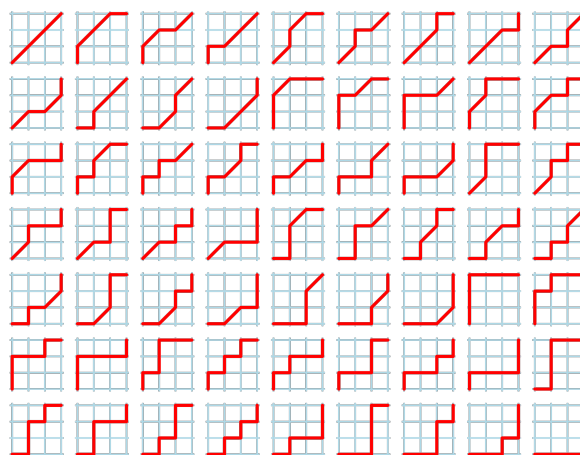


FIGURA 3. I 63 cammini di Delannoy sulla griglia 3×3 .

ESERCIZIO 7

Numeri di Delannoy.

1	1	1	1	1	1	1	1	1	1	1
1	3	5	7	9	11	13	15	17	19	
1	5	13	25	41	61	85	113	145	181	
1	7	25	63	129	231	377	575	833	1159	
1	9	41	129	321	681	1289	2241	3649	5641	
1	11	61	231	681	1683	3653	7183	13073	22363	
1	13	85	377	1289	3653	8989	19825	40081	75517	
1	15	113	575	2241	7183	19825	48639	108545	224143	
1	17	145	833	3649	13073	40081	108545	265729	598417	
1	19	181	1159	5641	22363	75517	224143	598417	1462563	

FIGURA 4. I numeri di Delannoy $D(i, j)$ con $i, j \in \{0, 1, \dots, 9\}$.

Tempo: 20 min.

Scrivete un programma ricorsivo che legga in ingresso due interi $m, n \geq 0$ e calcoli e visualizzi il numero di Delannoy $D(m, n)$. Nel caso in cui l'utente inserisca valori che non soddisfano la condizione $m, n \geq 0$, forzate il reinserimento.

ESERCIZIO 8

Tabulazione dei numeri di Delannoy.

Tempo: 25 min.

Scrivete un programma che legga in ingresso due interi $m, n \geq 0$ e calcoli e visualizzi una tabella rettangolare di dimensioni $m \times n$ il cui elemento di posto (i, j) sia il numero di Delannoy $D(i, j)$. Per esempio, nel caso in cui $m = n = 9$ l'output del vostro programma dovrebbe collimare con quello mostrato in Figura 4. Riutilizzate la funzione ricorsiva scritta per risolvere l'Esercizio 7.

ESERCIZIO 9

Successioni maschio-femmina di Hofstadter.

Tempo: 30 min.

Si consideri la coppia di successioni definite da

$$F(n) = n - M(F(n-1)), \quad (3)$$

$$M(n) = n - F(M(n-1)), \quad (4)$$

con valori iniziali

$$F(0) = 1, \quad (5)$$

$$M(0) = 0. \quad (6)$$

Si tratta delle *successioni maschio-femmina di Hofstadter*. Per maggiori informazioni si veda, ad esempio,

<http://mathworld.wolfram.com/HofstadterMale-FemaleSequences.html>

I primi dieci numeri della sequenza maschio $M(n)$ sono 0, 0, 1, 2, 2, 3, 4, 4, 5, 6. I primi dieci numeri della sequenza femmina $F(n)$ sono 1, 1, 2, 2, 3, 3, 4, 5, 5, 6.

Scopo di questo esercizio è il calcolo dei valori delle successioni maschio-femmina.

SPECIFICHE.

Si implementi, secondo le seguenti specifiche, un programma C per calcolare i primi $n + 1$ numeri delle sequenze $F(n)$ e $M(n)$, per un valore n inserito dall'utente.

- (a) Il programma chiede all'utente di inserire un numero intero $n \geq 0$, verificando la correttezza dell'input. Se $n < 0$ il programma forza il reinserimento.
- (b) Il programma visualizza in output i primi $n + 1$ numeri delle sequenze $F(n)$ e $M(n)$. Ad esempio, per $n = 4$ l'output del programma sarà:

n	F(n)	M(n)
0	1	0
1	1	0
2	2	1
3	2	2
4	3	2

(V. Marra) DIPARTIMENTO DI MATEMATICA *Federigo Enriques*, UNIVERSITÀ DEGLI STUDI DI MILANO, VIA CESARE SALDINI, 50, I-20133 MILANO

Email address: `vincenzo.marra@unimi.it`