

Politecnico di Milano

Informatica B, AA 2021/2022

Laboratorio 4

Luca Frittoli (luca.frittoli@polimi.it)
Mirko Salaris (mirko.salaris@polimi.it)

1 Dicembre 2021

1. **Riscaldamento** Costruisci le seguenti variabili senza fare utilizzo di cicli (e senza scriverle “manualmente”).

a =

1	2	3	4	5
---	---	---	---	---

b =

50	40	30	20	10
----	----	----	----	----

c =

51	42	33	24	15
----	----	----	----	----

mat1 =

0	0	0	0	0
0	0	0	0	0
0	0	0	0	0
0	0	0	0	0
0	0	0	0	57

mat2 =

1	2	3	4	5
2	4	6	8	10
3	6	9	12	15
4	8	12	16	20
5	10	15	20	25

mat3 =

1	2	3	4	5
1	4	9	16	25
1	8	27	64	125
1	16	81	256	625
1	32	243	1024	3125

2. **Acquisizione da tastiera** Si scriva un programma che prenda in input una sequenza di al più 10 numeri pari, e ne stampi la media. L'acquisizione dovrà terminare se viene inserito un numero dispari.
3. **Unità imperiali** Si scriva un programma che prenda in input una capacità espressa in galloni (gal) e pinte (pt) e la converta in litri (l). Conversione: 1 gal = 8 pt, 1 pt = 0.568 l.
4. **Polinomi** Scrivi un programma per calcolare il valore di un polinomio in un dato punto. Il programma chiede all'utente i coefficienti del polinomio e successivamente il valore della variabile x , e calcola la y corrispondente.

Bonus: stampa il grafico del polinomio nell'intervallo $[-5, 5]$, evidenziando il punto (x, y) calcolato in precedenza.

Suggerimento: per i coefficienti del polinomio è possibile utilizzare la funzione `input()` una sola volta; l'utente dovrà inserire i coefficienti tra parentesi quadre e separati da spazi.

Esempio: a input dell'utente "[1 0 3 4]" e successivamente 2 come valore per la x , corrisponderà il polinomio $x^0 + 3x^2 + 4x^3$ e quindi il risultato 45.

5. **Shift a sinistra** Si scriva un programma che, dato un array di cifre binarie, effettui lo shift a sinistra fino a quando non si ottiene un 1 nella prima posizione.

Esempio: [0 0 1 1 0 1 0] \rightarrow [1 1 0 1 0 0 0].

6. **Matrici simmetriche** Si scriva un programma che prenda in input una matrice quadrata e verifichi se è simmetrica o antisimmetrica.

Suggerimento: una matrice quadrata A è simmetrica se $A(k, l) = A(l, k)$ per ogni coppia di indici k, l , mentre è antisimmetrica se $A(k, l) = -A(l, k)$ per ogni coppia di indici k, l . Per controllare se due matrici sono uguali è possibile utilizzare la funzione `isequal`.

Bonus: si risolva l'esercizio senza utilizzare cicli.

7. **Approssimazione di π** Si scriva un programma che calcoli un'approssimazione di π usando la serie di Gregory-Leibniz:

$$\frac{\pi}{4} = 1 - \frac{1}{3} + \frac{1}{5} - \frac{1}{7} + \dots + \frac{(-1)^n}{2n+1} + \dots$$

Il programma dovrà prendere in input l'indice N a cui troncare la somma e stampare l'errore assoluto commesso nell'approssimazione.

Suggerimento: Matlab dispone di tantissime costanti matematiche come π (`pi`).

8. **Crivello di Eratostene** Implementare il Crivello di Eratostene, un semplice algoritmo per trovare tutti i numeri primi entro un certo N inserito dall'utente. L'algoritmo prende in input un array, detto "setaccio", contenente i numeri interi da 2 a N (1 non è un numero primo!). L'algoritmo cancella dall'array tutti i multipli del primo numero del setaccio (escluso il primo multiplo), poi tutti i multipli del secondo numero (escluso il primo multiplo), e così via, finché non rimangono solo i numeri primi.

Suggerimento 1: un modo per cancellare certi elementi di un array è cambiare il loro valore in `[]`.

Suggerimento 2: non è necessario scorrere tutti i numeri del setaccio, bastano quelli il cui quadrato non supera N . Infatti, se $p < N$ è un numero primo tale che $p^2 > N$, i suoi multipli $p \cdot r$ con $r < p$ sono già stati cancellati dall'algoritmo, quindi il minimo multiplo non ancora cancellato sarebbe p^2 , che però è già fuori dal setaccio in quanto $p^2 > N$.

9. **Sottomatrici** Si scriva un programma che prenda in input due matrici A e B e controlli se B è una sottomatrice di A .

10. **Numeri di Fibonacci** Si scriva una funzione `fibonacci` che prenda in input un intero positivo n e calcoli il corrispondente elemento nella successione di Fibonacci $0, 1, 1, 2, 3, 5, 8, 13, \dots$. Si testi la funzione utilizzando il seguente script:

```
n = randi([1,20], 10);
passed_test = 0;
for i=1:10
    if fibonacci(n(i)) + fibonacci(n(i)+1) == fibonacci(n(i)+2)
        passed_test = passed_test + 1;
    else
        disp(['did not pass test ', num2str(i)])
    end
end

disp(['Passed ', num2str(passed_test), ' out of 10'])
```

Suggerimento: si ricordi che ogni elemento della successione di Fibonacci (eccetto i primi due, che sono per definizione $0, 1$) è la somma dei due precedenti.