



POLITECNICO
MILANO 1863

*Esercitazione 3: Funzioni logiche
e immagini*

Ing. Andrea Mazzoleni

P1: Valore più vicino

Scrivere una funzione *closestVal* che prende in ingresso un vettore **vett** ed uno scalare **x** e restituisce il valore di **vett** più vicino ad **x** e la sua posizione.

a

```
function [closest, pos_closest] = closestVal(vett, val)
```

Es

```
[closest, pos_closest] = closestVal([100 200 20 45 70 10], 33)
```

```
closest=45
```

```
pos_closest=4
```

P3: Palindroma

Scrivere un programma che determina se una parola è palindroma sfruttando le funzioni logiche.

256 codici per codifica dei caratteri (%c).

- Per passare dal carattere al codice
`codice_ascii = double(carattere)`
- Viceversa
`Carattere = char(codice_ascii)`

Codici alfabeto:

- a-z da 97 a 122
 - A-Z da 65 a 90
- `double('A') ~ = double('a')`

P3: Anagramma

Scrivere un programma che richiede in ingresso due parole e determina se l'una è l'anagramma dell'altra

- Uno script si occupa dell'acquisizione delle parole.
- Implementare una funzione per creare l'istogramma delle parole.
- Eseguire nello script il confronto tra istogrammi e visualizzare a schermo il risultato. (Usare funzioni logiche)

P4 - Unione e intersezione

Scrivere un programma che richiede in ingresso due array A e B di dimensioni non necessariamente uguali:

- Scrivere una funzione che ricevuti A e B calcoli il vettore intersezione tra i due
- Scrivere una funzione che ricevuti A e B calcoli il vettore unione tra i due

P.S. Si gestisca anche il caso in cui ci siano doppioni nello stesso vettore.