

Elementi di Informatica e Applicazioni Numeriche T

Esercizio: Divisioni Ripetute

Esercizio: Divisioni Ripetute

Per ottenere la rappresentazione binaria di un intero n si usa:

```
function D = dec2bin(n)
```

- Funziona solo se n non ha parte frazionaria
- Restituisce una stringa di 0/1

Nel file di funzione `es_dec2bin`, si definisca la funzione ausiliaria:

```
function v = my_dec2bin(n)
```

- Che faccia lo stesso, utilizzando il metodo delle divisioni ripetute
- Per semplicità, si restituisca però un vettore di 0/1

Esercizio: Divisioni Ripetute

Si verifichi la correttezza:

- Nella funzione principale `es_dec2bin`
- Confrontandosi (visivamente) con la funzione di Matlab

Alcune informazioni utili:

- La divisione intera tra numeri positivi potete farla...
- ...Arrotondando per difetto il risultato di una divisione normale!

```
floor(<dividendo> ./ <divisore>)
```

- Il resto della divisione intera si ottiene con:

```
mod(<dividendo>, <divisore>)
```

Elementi di Informatica e Applicazioni Numeriche T

Esercizio: Trasposizione di Matrice

Esercizio: Trasposizione di Matrice

Nel file di funzione `es_transpose` si definisca la funzione ausiliaria:

```
function B = my_transpose(A)
```

- Che calcoli la trasposizione della matrice **A**

Si verifichi la correttezza:

- Nella funzione principale `es_transpose`
- Utilizzando delle matrici di numeri casuali
- Confrontandosi con l'operatore di trasposizione di Matlab

Elementi di Informatica e Applicazioni Numeriche T

Esercizio: Conteggio di Elementi

Esercizio: Conteggio di Elementi

Nel file di funzione `es_count.m`, si definisca la funzione ausiliaria:

```
function [U, C] = my_count(V)
```

Che, dato un vettore di ingresso \mathbf{v} , restituisca \mathbf{U} e \mathbf{C} tali che:

- \mathbf{U} contenga gli elementi distinti di \mathbf{v}
- Per ogni \mathbf{v} in \mathbf{U} , \mathbf{C} contenga il numero di occorrenze di \mathbf{v} in \mathbf{U}
- I.e. la funzione deve contare le occorrenze di ogni elemento.

Si verifichi il funzionamento nella funzione principale

- Si utilizzi un vettore definito a mano

Suggerimento: la funzione `unique` restituisce gli elementi distinti

Elementi di Informatica e Applicazioni Numeriche T

Esercizio: Prodotto Matriciale

Esercizio: Prodotto Matriciale

Nel file di funzione `es_mprod`, si definisca la funzione ausiliaria:

Si definisca una funzione:

```
function Z = my_mprod(A, B)
```

- Che calcoli il prodotto matriciale di **A** e **B**

Si verifichi la correttezza:

- Nella funzione principale
- Utilizzando matrici di numeri casuali (attenzione alle dimensioni!)
- Confrontandosi con l'operatore di prodotto in Matlab

Suggerimento: ogni prodotto riga/colonna è un prodotto scalare!

Elementi di Informatica e Applicazioni Numeriche T

Esercizio: Prodotto Cumulativo

Esercizio: Prodotto Cumulativo

Matlab fornisce la funzione:

```
function P = cumprod(V)
```

- Che in ogni elemento $P(ii)$ del vettore restituito...
- ...riporta il prodotto degli elementi di v negli indici da 1 a ii

Quindi, per esempio:

```
cumprod([2, 4, 6]) % denota [2, 8=2*4, 48=2*4*6]
```

Esercizio: Prodotto Cumulativo

Nel file di funzione `es_cumprod.m`, si definisca una funzione:

```
function P = my_cumprod(V)
```

- Che replichi il comportamento di `cumprod`

Si verifichi la correttezza nella funzione principale:

- Si utilizzino dei vettori di numeri casuali
- Si confrontino i risultati con quelli di `cumprod` in Matlab