



POLITECNICO
DI MILANO

Lab 6 – Info B



Luca M. Cassano - luca.cassano@polimi.it

Sadegh M. Astaneh - sadegh.astaneh@unimi.it

Matteo Papini - matteo.papini@polimi.it



Introduzione a Matlab/Octave

Grafici

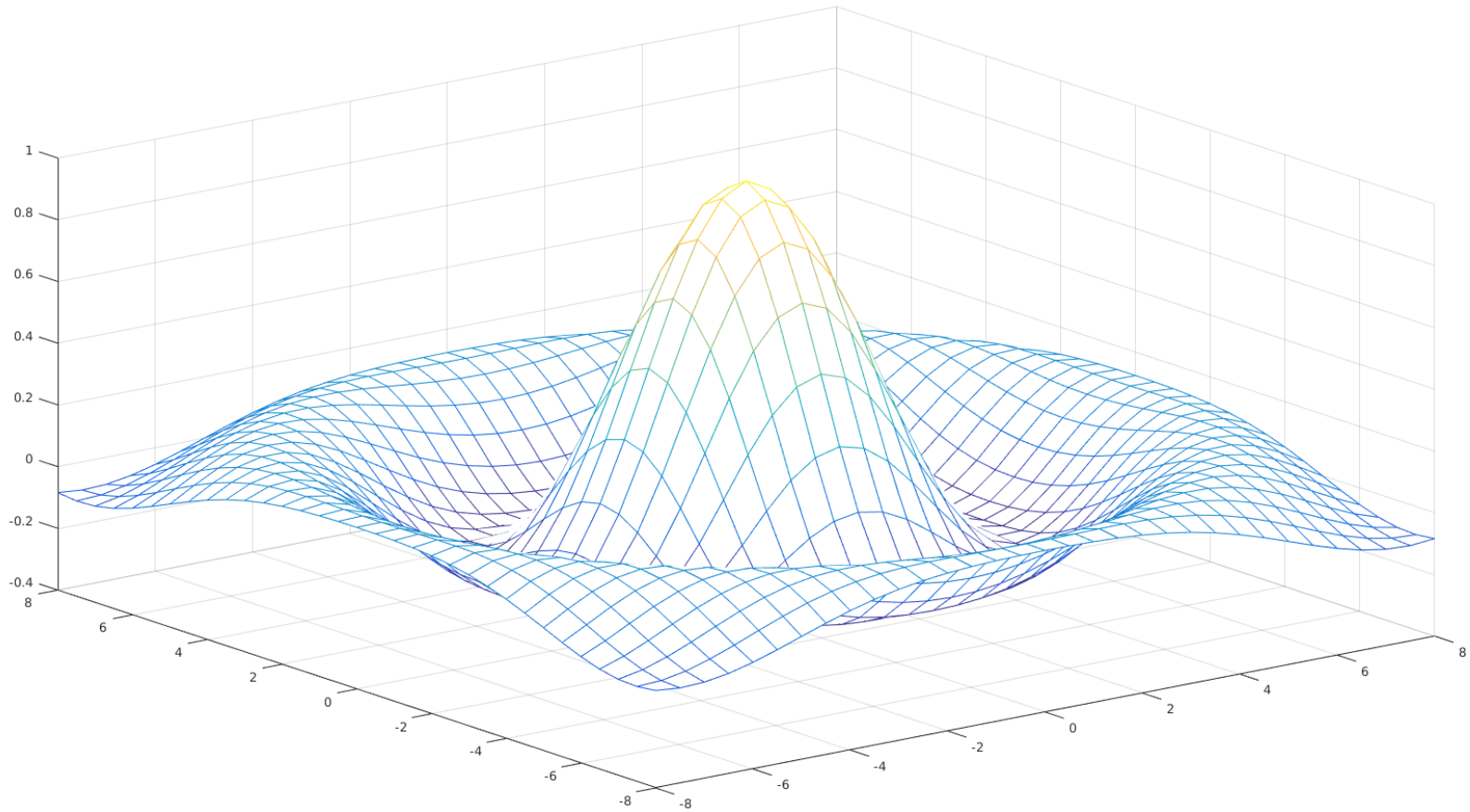
Funzioni Ricorsive

Strutture

- Superficie di rotazione
- Disegno 2D
- Mappamondo

Scrivere un programma che permetta di visualizzare la funzione bidimensionale $\sin(r)/r$, lungo le due direzioni x e y

N.B.: r è la distanza dall'origine nel piano XY



Scrivere un programma di disegno 2D interattivo:

- L'utente può scegliere se disegnare un segmento o un cerchio
- Segmento: l'utente fornisce i due estremi come array di Matlab (e.g. $[-1,-1]$ e $[2,3]$)
- Cerchio: l'utente fornisce centro e raggio
- L'utente sceglie se disegnare altre figure (segmenti e cerchi) o smettere
- Tutte le figure devono essere disegnate sullo stesso piano

Equazione parametrica del segmento:

Dati gli estremi (x_1, y_1) e (x_2, y_2) :

$$\begin{cases} t \in [0, 1] \\ x(t) = x_1 * t + x_2 * (1 - t) \\ y(t) = y_1 * t + y_2 * (1 - t) \end{cases}$$

Equazione parametrica del cerchio:

Dato il centro (x_0, y_0) e il raggio r :

$$\begin{cases} t \in [0, 2\pi] \\ x(t) = x_0 + r * \cos(t) \\ y(t) = y_0 + r * \sin(t) \end{cases}$$

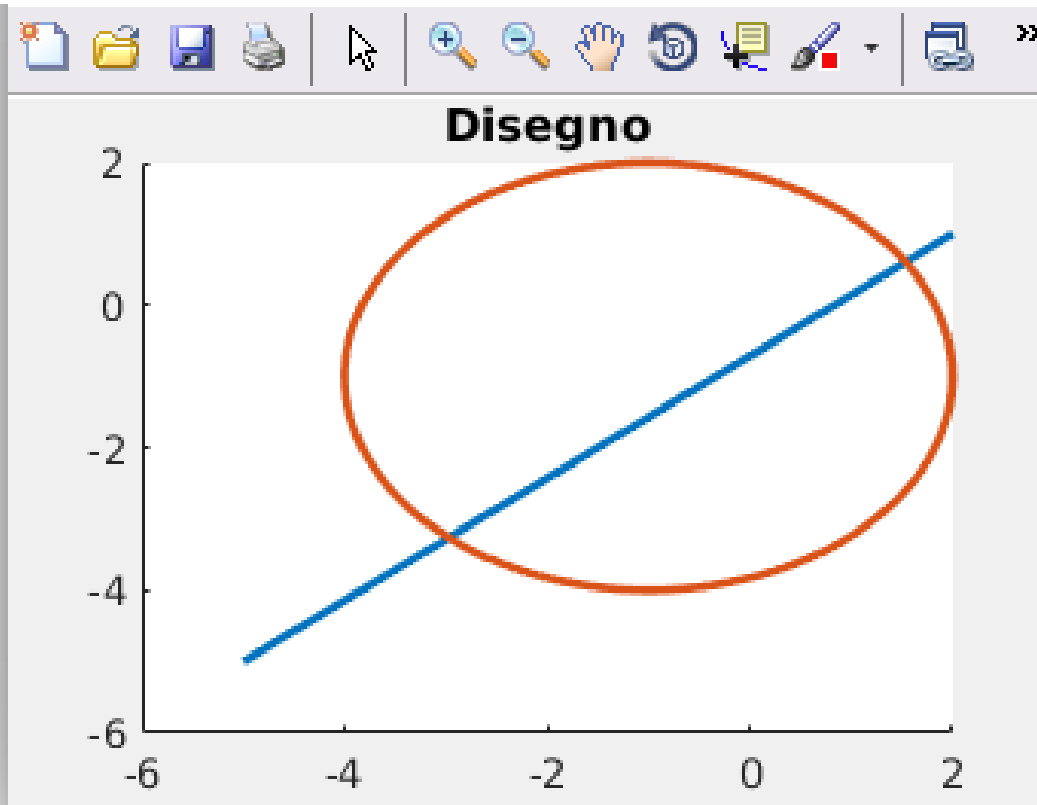


Esempio



```
Cosa vuoi disegnare?  
segmento  
Inserire primo punto  
[-5,-5]  
Inserire secondo punto  
[2,1]  
Cosa vuoi disegnare?  
cerchio  
Inserisci centro  
[-1,-1]  
Inserisci raggio  
3  
Cosa vuoi disegnare?  
esci
```

```
fx >>
```



Scrivere un programma che permetta di visualizzare località sul mappamondo date le coordinate geografiche

- Rappresentare il mappamondo con una sfera di raggio unitario
- Visualizzare sulla sfera l'equatore e il meridiano di Greenwich
- In modo interattivo, l'utente può inserire latitudine (da -90° a 90°) e longitudine (da -180° a 180°). Visualizzare il punto corrispondente sulla sfera

Un po' di geografia

Latitudine: da -90° (Sud) a 90° (Nord)

Longitudine: da -180° (Ovest) a 180° (Est)

Equatore: cerchio di latitudine 0

Meridiano di Greenwich: **semicerchio** di
longitudine 0

Equazione parametrica del cerchio:

Dato il centro (x_0, y_0) e il raggio r :

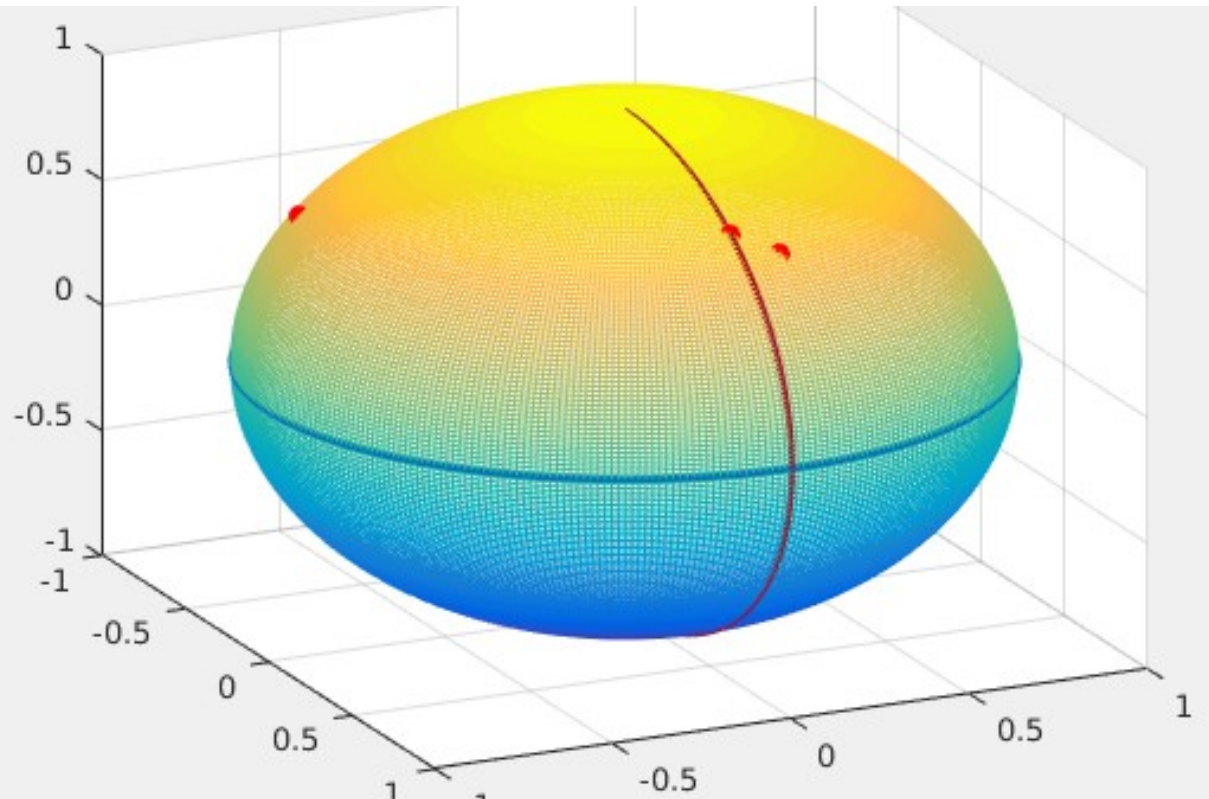
$$\begin{cases} t \in [0, 2\pi] \\ x(t) = x_0 + r * \cos(t) \\ y(t) = y_0 + r * \sin(t) \end{cases}$$

Equazione parametrica della sfera:

Dato il centro (x_0, y_0, z_0) e il raggio r :

$$\begin{cases} t_1, t_2 \in [0, 2\pi] \\ x(t) = x_0 + r * \cos(t_1) * \cos(t_2) \\ y(t) = y_0 + r * \cos(t_1) * \sin(t_2) \\ z(t) = z_0 + r * \sin(t_1) \end{cases}$$

```
Inserire latitudine  
45  
Inserire longitudine  
9  
Vuoi disegnare ancora?  
si  
Inserire latitudine  
51  
Inserire longitudine  
0  
Vuoi disegnare ancora?  
si  
Inserire latitudine  
34  
Inserire longitudine  
-118  
Vuoi disegnare ancora?  
no  
fx >>
```



Scrivere un programma che chieda all'utente di inserire una serie di dati contenenti ognuno i seguenti attributi:

- città (stringa)
- giorno (intero positivo)
- mese (intero positivo)
- anno (intero positivo)
- tipo di misurazione (char)
- valore (reale)

Ad esempio, l'utente potrà inserire:

- Milano
- 04
- 12
- 2017
- N
- 10.5

- Dopo aver acquisito una certa quantità di dati, il programma dovrà chiedere all'utente il nome di una città e un tipo di misurazione.
- A questo punto il programma cercherà nell'archivio tutti i record riguardanti la città e il tipo di misurazione richiesti. Stamperà poi a video i dati selezionati ed il relativo valore minimo, massimo e medio dei valori.



- I numeri di Fibonacci (dinamiche di popolazione)
- Il Massimo Comun Divisore (algoritmo di Euclide)
- Il problema delle torri di Hanoi

I numeri di Fibonacci



Idea di base

1) $\text{fib}(n)=1$ se $n=0$ oppure $n=1$

2) $\text{fib}(n)= \text{fib}(n-1) + \text{fib}(n-2)$ se $n>1$

Definizione:

1) $\text{MCD}(m,n) = m$ se $m = n$

2a) $\text{MCD}(m,n) = \text{MCD}(m-n,n)$ se $m > n$

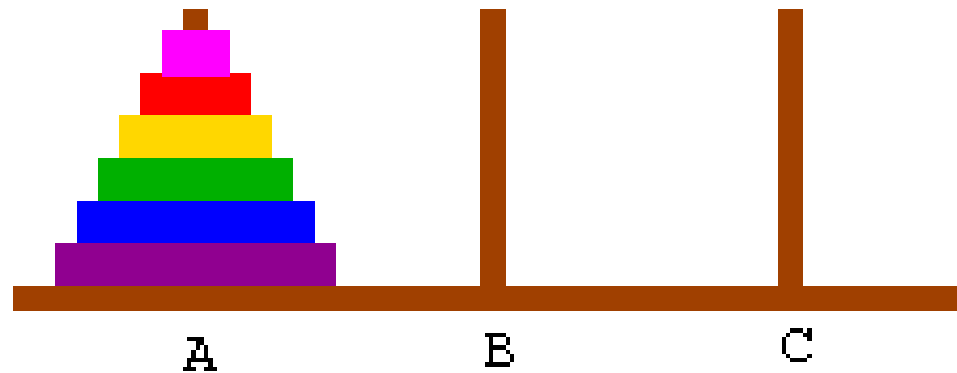
2b) $\text{MCD}(m,n) = \text{MCD}(m,n-m)$ se $n > m$

esempio:

$$\begin{aligned} \text{MCD}(21,56) &= \text{MCD}(21,35) = \\ &\text{MCD}(21,14) = \\ &= \text{MCD}(7,14) = \text{MCD}(7,7) = 7 \end{aligned}$$

Le torri di Hanoi

<http://www.cs.cmu.edu/~cburch/survey/recurse/hanoi.html>



Problema: spostare tutti i dischi dalla torre A alla torre B (usando la torre C come “supporto intermedio”) in modo che si trovino nello stesso ordine



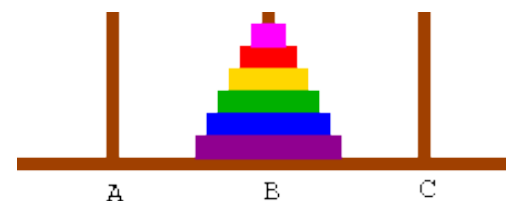
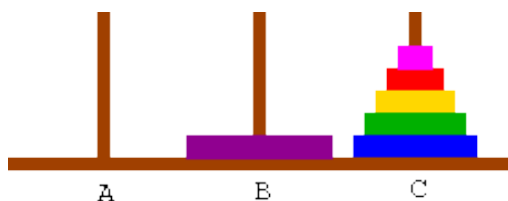
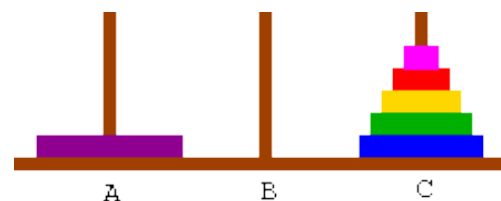
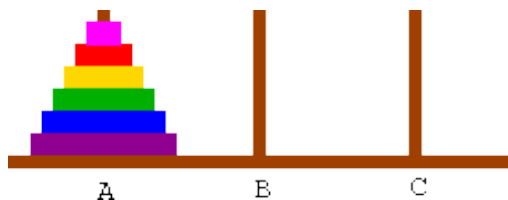
- Scriveremo una funzione ricorsiva che prende come parametro il numero del disco più grande che vogliamo spostare (da 0 a 5 come nel disegno)
- La funzione prenderà anche tre parametri che indicano:
 - da quale asta vogliamo partire (source),
 - a quale asta vogliamo arrivare (dest),
 - l'altra asta, che possiamo usare come supporto temporaneo (spare).

- Voglio spostare n anelli dal piolo sorgente, a quello destinazione, usando come appoggio il piolo ausiliario
 - Devo quindi prima spostare $n - 1$ anelli dal sorgente all'ausiliario, usando come appoggio il piolo destinazione
 - Poi sposto l'unico anello rimasto dal sorgente al piolo destinazione
 - Infine sposto gli $n - 1$ anelli che si trovano sull'ausiliario all'anello destinazione..

- Quando si spostano gli $n - 1$ anelli la funzione hanoi richiama se stessa, cioè effettua una chiamata ricorsiva, semplificando però il problema perché bisogna spostare un numero di anelli inferiore.
- In pratica, con la ricorsione il problema viene continuamente ridotto di complessità fino alla soluzione banale in cui rimane solo un anello, che viene semplicemente spostato nel piolo destinazione.



Le torri di Hanoi: strategia



Ridurremo il problema a quello di spostare 5 dischi dalla torre C alla torre B, dopo che il disco 5 è stato già messo nella posizione giusta



```
FUNCTION MoveTower(disk, source, dest, spare):
IF disk == 0, THEN:
    move disk from source to dest
ELSE:
    MoveTower(disk - 1, source, spare, dest)
    /* (Passo 1) */
    move disk from source to dest //
    /* (Passo 2) */
    MoveTower(disk - 1, spare, dest, source) //
    /* (Passo 3) */
END IF
```

Nota: l'algoritmo aggiunge un caso base: quando il disco è il più piccolo (il numero 0). In questo caso possiamo muoverlo direttamente perché non ne ha altri sopra. Negli altri casi, seguiamo la procedura descritta per il disco 5.