

Università degli Studi di Trieste

Corsi di ingegneria industriale e navale

Esercitazioni di Fondamenti di Informatica

Giacomo Strangolino

mailto/chat: delleceste@gmail.com

<http://www.giacomos.it>

Lezione 4 (23/11/2011)

- Scrivere una funzione che, data la matrice completa che rappresenta un sistema lineare, ne trovi l'insieme delle soluzioni con il metodo di Gauss-Jordan.
- Si supponga che la matrice fornita rappresenti già un sistema determinato

Lezione 4

Un sistema lineare di m equazioni in m incognite

$$\begin{cases} a_{1,1}x_1 + a_{1,2}x_2 + \cdots + a_{1,n}x_n &= b_1 \\ a_{2,1}x_1 + a_{2,2}x_2 + \cdots + a_{2,n}x_n &= b_2 \\ &\vdots \\ a_{m,1}x_1 + a_{m,2}x_2 + \cdots + a_{m,n}x_n &= b_m \end{cases}$$

Può essere descritto dalla sua **matrice completa dei coefficienti** del sistema:

$$\begin{pmatrix} a_{1,1} & \cdots & a_{1,n} & b_1 \\ \vdots & \ddots & \vdots & \vdots \\ a_{m,1} & \cdots & a_{m,n} & b_m \end{pmatrix}$$

Per il nostro esercizio supponiamo che i coefficienti del sistema lineare appartengano a \mathbf{R} .
Se con \mathbf{A} indico la matrice dei coefficienti, con \mathbf{B} il vettore colonna dei termini noti e con \mathbf{x} il vettore colonna delle incognite, allora il sistema si può scrivere come

$$\mathbf{Ax} = \mathbf{B}$$

Lezione 4

Per risolvere un sistema lineare si può:

- scambiare l'ordine di scrittura di due equazioni;
- moltiplicare entrambi i membri di un'equazione per un numero diverso da zero;
- sommare ad ogni membro di un'equazione la stessa quantità a sinistra e a destra.

Lezione 4

Sulla matrice, ciò equivale a:

- scambiare tra loro 2 righe;
- moltiplicare una riga per uno scalare non nullo;
- sostituire una riga con quella ottenuta sommando ad essa un multiplo di un'altra riga.

Obiettivo:

- Ottenere una matrice A a diagonale unitaria;



Il vettore B conterrà allora le soluzioni del sistema

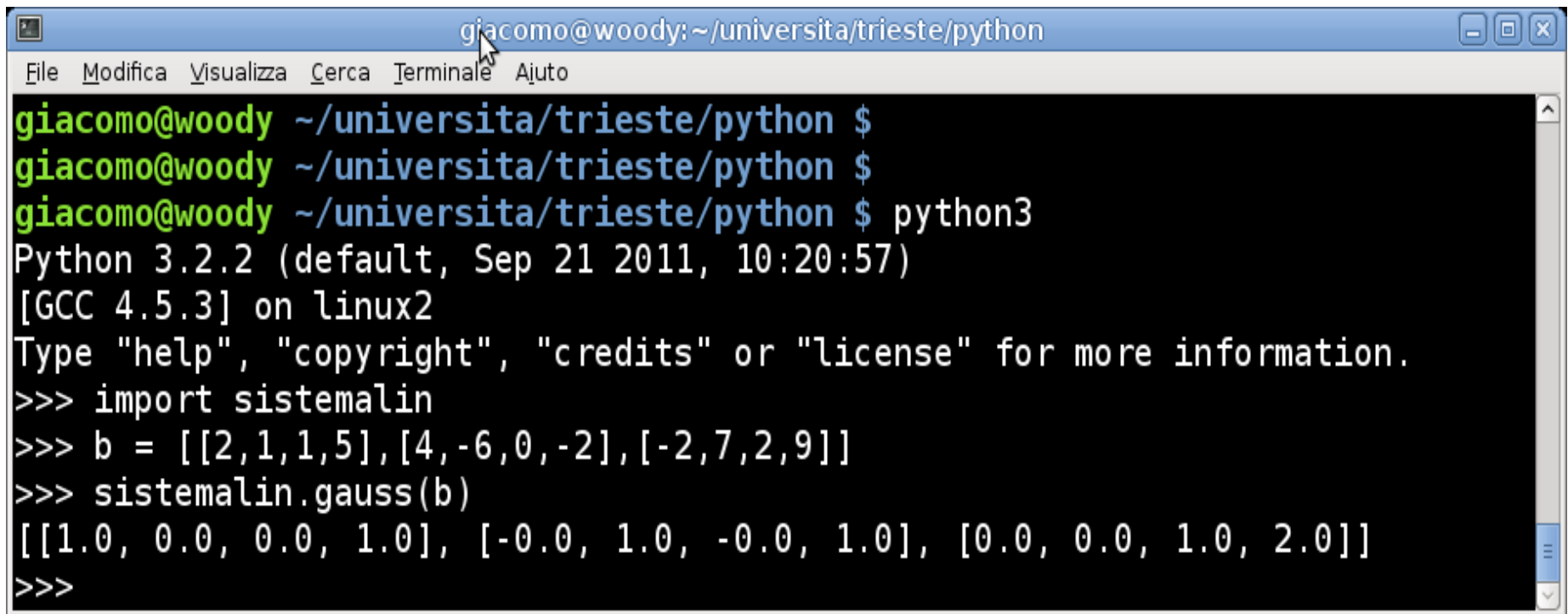
Lezione 4

$$\begin{cases} 2x + y + z = 5 \\ 4x - 6y = -2 \\ -2x + 7y + 2z = 9 \end{cases}$$

$$\begin{bmatrix} 2 & 1 & 1 & 5 \\ 4 & -6 & 0 & -2 \\ -2 & 7 & 2 & 9 \end{bmatrix} \rightarrow \begin{bmatrix} 2 & 1 & 1 & 5 \\ 0 & -8 & -2 & -12 \\ 0 & 8 & 3 & 14 \end{bmatrix} \rightarrow \begin{bmatrix} 2 & 1 & 1 & 5 \\ 0 & -8 & -2 & -12 \\ 0 & 0 & 1 & 2 \end{bmatrix}$$

E così via... annullando successivamente anche gli elementi sopra la diagonale e finalmente rendendo la diagonale unitaria

Lezione 4

A terminal window titled 'giacomo@woody:~/universita/trieste/python'. The window has a menu bar with 'File', 'Modifica', 'Visualizza', 'Cerca', 'Terminale', and 'Aiuto'. The terminal shows the following commands and output:

```
giacomo@woody ~/universita/trieste/python $  
giacomo@woody ~/universita/trieste/python $  
giacomo@woody ~/universita/trieste/python $ python3  
Python 3.2.2 (default, Sep 21 2011, 10:20:57)  
[GCC 4.5.3] on linux2  
Type "help", "copyright", "credits" or "license" for more information.  
>>> import sistemalin  
>>> b = [[2,1,1,5],[4,-6,0,-2],[-2,7,2,9]]  
>>> sistemalin.gauss(b)  
[[1.0, 0.0, 0.0, 1.0], [-0.0, 1.0, -0.0, 1.0], [0.0, 0.0, 1.0, 2.0]]  
>>>
```



Soluzioni: $x = 1.0$ $y = 1.0$ $z = 2.0$

Lezione 4

- Si scriva un programma che legga da tastiera un testo e stampi (in verticale) l'istogramma che rappresenta il numero di volte che ciascuna lettera è comparsa nel testo

- Esempio: “pippo”

*

*

iop

- Suggerimento: salvare asterischi in una matrice con numero di righe pari al massimo alla lunghezza stringa e numero di colonne pari al numero possibile di caratteri ASCII standard. Successivamente stampare la matrice all'incontrario (dall'alto verso il basso). Ad esempio ogni volta che trovo una 'p' salvo un '*' alla riga r pari alla somma delle 'p' trovate finora nella colonna `ord('p')`

-

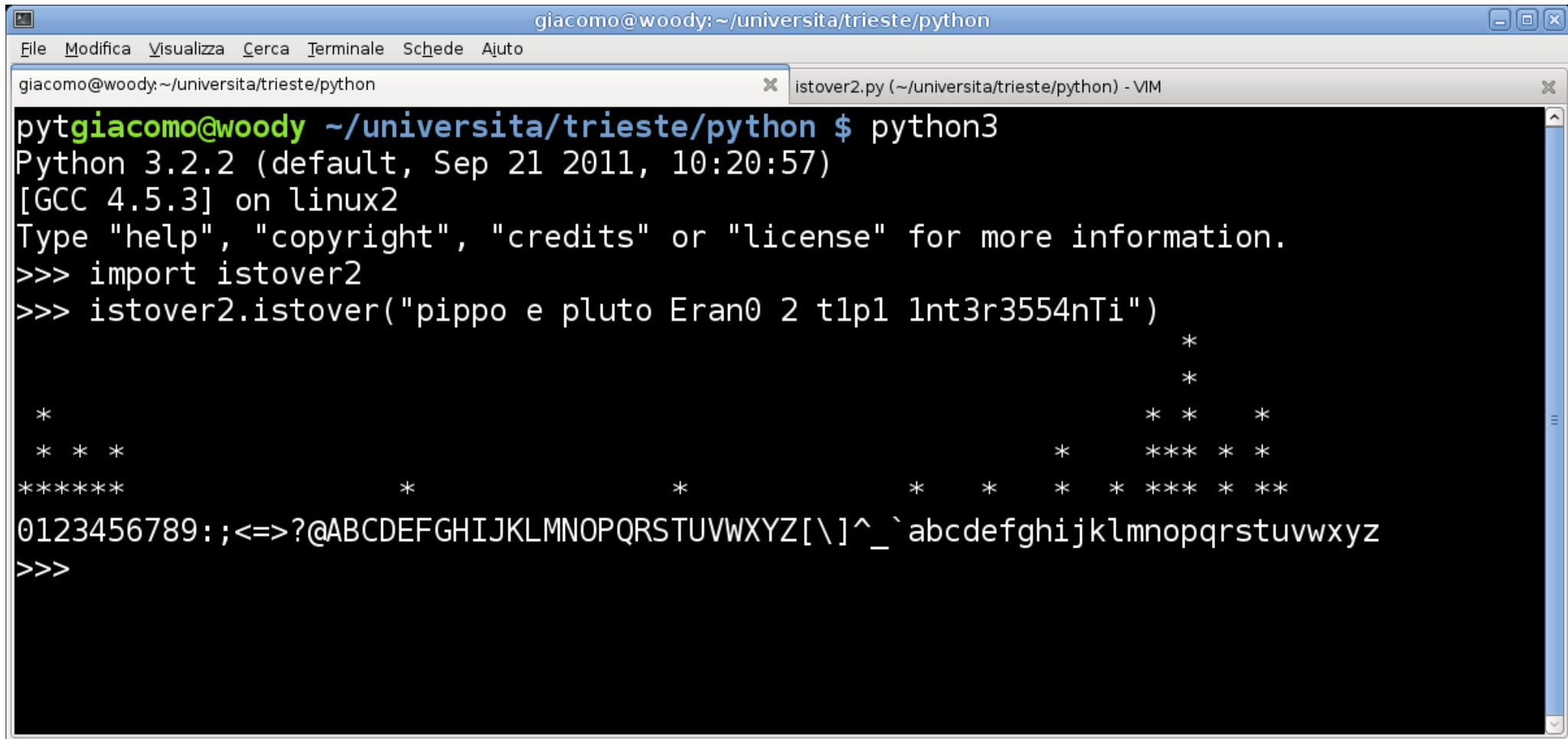
Lezione 4

Homework

- Riscrivere il programma *Istogramma Verticale* in modo che stampi un istogramma orizzontale;
- Prendendo spunto dall'esercizio sui sistemi lineari, scrivere una funzione che calcoli il determinante di una matrice
- Riscrivere il programma *Istogramma Verticale* in modo che vengano stampati sulle “ascisse” tutti i caratteri dallo '0' alla 'z' e gli asterischi solo in corrispondenza dei caratteri presenti nella stringa di input (vedi esempio slide successiva)

Lezione 4

Homework (istover)



The screenshot shows a terminal window titled "giacomo@woody: ~/universita/trieste/python". The window contains a Python 3.2.2 prompt where the user has imported a module named "istover2" and called the function "istover" with the argument "pippo e pluto Eran0 2 t1p1 1nt3r3554nTi". The output of the function is a large ASCII art graphic. The graphic consists of several lines of asterisks (*) and a long line of ASCII characters. The characters are arranged in a way that suggests a hidden message or a specific pattern. The characters include digits, punctuation, and letters, all in uppercase and lowercase. The overall shape of the graphic is roughly rectangular, with the characters filling the space between the asterisks.

```
giacomo@woody ~/universita/trieste/python $ python3
Python 3.2.2 (default, Sep 21 2011, 10:20:57)
[GCC 4.5.3] on linux2
Type "help", "copyright", "credits" or "license" for more information.
>>> import istover2
>>> istover2.istover("pippo e pluto Eran0 2 t1p1 1nt3r3554nTi")
*
*
* * *
* * *
*****
0123456789:;<=>?@ABCDEFGHIJKLMNOPQRSTUVWXYZ[\]^_`abcdefghijklmnopqrstuvwxyz
>>>
```

Si noti che lo spazio viene trascurato perché ha codice ASCII $32 < \text{ord}('0')$ (v. tabella slide seguente)

Lezioni 1 e 2

Tabella ASCII:

000	(nul)	016	► (dle)	032	sp	048	0	064	@	080	P	096	`	112	p
001	☉ (soh)	017	◄ (dc1)	033	!	049	1	065	A	081	Q	097	a	113	q
002	☼ (stx)	018	↕ (dc2)	034	"	050	2	066	B	082	R	098	b	114	r
003	♥ (etx)	019	≡ (dc3)	035	#	051	3	067	C	083	S	099	c	115	s
004	♦ (eot)	020	⌘ (dc4)	036	\$	052	4	068	D	084	T	100	d	116	t
005	♣ (enq)	021	§ (nak)	037	%	053	5	069	E	085	U	101	e	117	u
006	♠ (ack)	022	— (syn)	038	&	054	6	070	F	086	V	102	f	118	v
007	• (bel)	023	↕ (etb)	039	'	055	7	071	G	087	W	103	g	119	w
008	■ (bs)	024	↑ (can)	040	(056	8	072	H	088	X	104	h	120	x
009	(tab)	025	↓ (em)	041)	057	9	073	I	089	Y	105	i	121	y
010	(lf)	026	(eof)	042	*	058	:	074	J	090	Z	106	j	122	z
011	♂ (vt)	027	← (esc)	043	+	059	;	075	K	091	[107	k	123	{
012	♀ (np)	028	L (fs)	044	,	060	<	076	L	092	\	108	l	124	
013	(cr)	029	↔ (gs)	045	-	061	=	077	M	093]	109	m	125	}
014	♪ (so)	030	▲ (rs)	046	.	062	>	078	N	094	^	110	n	126	~
015	✱ (si)	031	▼ (us)	047	/	063	?	079	O	095	_	111	o	127	△