

Laboratorio di Calcolo per Fisici, Prima esercitazione

Canale Lp-P, Docente: Cristiano De Michele

Lo scopo della prima esercitazione di laboratorio è di introdurre gli strumenti di base che verranno usati nel corso delle successive esercitazioni: la *shell*, l'editor di testo, il compilatore *gcc*, e *python* (tramite le librerie *matplotlib* e *numpy*) per la grafica.

► Prima parte (obbligatoria)

1. Effettuare il *login* sulla propria macchina *Unix* utilizzando lo *userid* `lccdmxxx`, dove `xxx` è il numero del gruppo a cui siete stati assegnati (la password è identica allo *userid*).
2. Aprire una finestra di *terminale*.
3. Creare una cartella `EX1` con il comando `mkdir`. Tale cartella conterrà il materiale della prima esercitazione.
4. Nella cartella `EX1` aprire con l'editor di testo (consigliato: `emacs`) il file `temp.c`, e digitare il listato sottostante. Salvare il contenuto del file. *Suggerimento*: Per entrare nella cartella `EX1` usare il comando linux `cd`, e per aprire il file usare il comando `emacs temp.c`.

```
1 #include <stdio.h>
2
3 int main()
4 {
5     double tc, tf, conv, offset;
6     conv = 5./9.;
7     offset = 32.;
8     printf("Inserisci la temperatura in gradi Celsius \n");
9     scanf("%lf", &tc);
10    tf = tc/conv + offset;
11    printf("La temperatura in gradi Fahrenheit vale %5.2f gradi\n",tf);
12 }
```

Listato 1: Programma `temp.c`

5. Compilare il programma in `c` digitando sul terminale: `gcc temp.c -o temp.x`
6. Eseguire il file `temp.x` digitando `./temp.x`
7. Inserire i dati richiesti dal programma; il programma è un semplice convertitore di temperature da gradi Celsius a gradi Fahrenheit.

► Seconda parte (obbligatoria)

1. Eseguire il programma `temp.x` quattro o più volte, con valori di input diversi, e creare un file di testo chiamato `temp.dat` con due colonne, che contenga i valori delle temperature in Celsius (`Tc`) e in Fahrenheit (`Tf`), cioè i valori di input e output del programma `temp.c`.

2. Aprire con l'editor di testo il file `ex1_2.py` e digitare il listato contenuto sottostante rispettando accuratamente l'indentazione. Il file `ex1_2.py` contiene un semplice script in linguaggio python per creare un plot (grafico) nel piano cartesiano, utilizzando i dati contenuti in un file con due colonne x, y .

```
1 import matplotlib.pyplot as plt
2 import numpy as np
3 plt.title('Un primo plot con Python')
4 x, y = np.loadtxt('temp.dat', unpack=True)
5 plt.plot(x, y, 'x', label='Temperature caricate da file')
6 plt.xlim((-10,50)) # intervallo lungo asse x
7 plt.ylim((10,125)) # intervallo lungo asse y
8 plt.show()
```

Listato 2: Programma *python* `ex1_2.py`

3. Dal terminale graficare i dati contenuti nel file `temp.dat` con il comando:

```
python3 ./ex1_2.py
```

4. Aggiungere le legende all'asse x e y aprendo il file `ex1_2.py` con l'editor di testo e inserendo prima di `plt.show()` i comandi:
`plt.xlabel('Tc')`
`plt.ylabel('Tf')`
5. Eseguire nuovamente lo script.
6. Aggiungere prima di `plt.show()` il seguente comando:
`plt.savefig('temp.png')`

7. Eseguire nuovamente lo script. Impartendo il comando `'ls'` nel terminale troverete che è stato creato il file `'temp.png'` che è un'immagine png del plot ottenuto con python. Per visualizzarlo potete usare il comando:

```
xdg-open ./temp.png
```

8. Notate che potete personalizzare colori, simboli e tipo di linea come indicato in figura 1. Ad esempio, sostituendo la stringa `'x'` nel comando `plt.plot()` con `'o--g'`, otterreste una linea verde tratteggiata dove i vari punti saranno rappresentati con dei cerchietti. Fate questa modifica ed eseguite nuovamente lo script python. Provate anche altri possibili scelte.

► Terza parte (obbligatoria)

Con *Python* e le librerie *matplotlib* è possibile graficare non solo dati contenuti in un file esterno, ma anche funzioni definite dall'utente. Per esempio, per plottare la funzione $y = x$ è sufficiente aggiungere prima di `plt.title`:

```

1 x = np.linspace(-10, 50, 100)
2 y = x
3 plt.plot(x, y, label='retta y=x')

```

In questo modo viene aggiunto il plot della retta $y = x$ nell'intervallo $[-10, 50]$.

1. Utilizzando le funzionalità in *python* appena descritte, disegnare la retta che interpola i dati generati dal programma `temp.x`.
2. Inserire una legenda nel grafico per la retta interpolante aggiungendo nel programma *python* prima di `plt.show()` il comando `plt.legend()` e riscalare gli assi x e y in modo che vadano dalla più piccola alla più grande delle temperature scelte. *Suggerimento:* Per modificare il range degli assi x e y modificare gli argomenti di `plt.xlim()` e `plt.ylim()`.

marker options

'.'	point marker	's'	square marker
','	pixel marker	'p'	pentagon marker
'o'	circle marker	'*'	star marker
'v'	triangle_down marker	'h'	hexagon1 marker
'^'	triangle_up marker	'H'	hexagon2 marker
'<'	triangle_left marker	'+'	plus marker
'>'	triangle_right marker	'x'	x marker
'1'	tri_down marker	'D'	diamond marker
'2'	tri_up marker	'd'	thin_diamond marker
'3'	tri_left marker	' '	vline marker
'4'	tri_right marker	'_'	hline marker

linestyle options

'-'	solid line style
'--'	dashed line style
'-.'	dash-dot line style
'...'	dotted line style

color options

'b'	blue
'g'	green
'r'	red
'c'	cyan
'm'	magenta
'y'	yellow
'k'	black
'w'	white

Figura 1: opzioni per personalizzare lo stile di un grafico