

Programmazione e Algoritmi 1 - Prova pratica

A.A. 2025/26 — Compito A del 15/1/2026

prof. Gianluca Amato

Esercizio 1 (8 punti)

Usando la libreria `ezgraphics`, scrivere un programma che disegna una spezzata (ovvero una sequenza di segmenti collegati l'uno all'altro) seguendo le istruzioni contenute nel file `disegno1.txt`. Il file in questione è composto da due righe, ognuna delle quali contiene una sequenza di numeri interi separati da spazi. Queste due righe rappresentano le coordinate di una serie di punti: in particolare, la prima riga contiene le coordinate x dei punti, mentre la seconda riga contiene le coordinate y . Il compito del vostro programma è disegnare una spezzata unendo i punti indicati nel file.

Ad esempio, supponiamo che il file `disegno1.txt` contenga quanto segue:

```
100 150 200 250 300
100 100 50 100 100
```

Il programma deve tracciare una linea dal punto di coordinate (100, 100) al punto (150, 100), poi una linea dal punto (150, 100) al punto (200, 50), poi una linea dal punto (200, 50) al punto (250, 100) e infine dal punto (250, 100) a (300, 100). In particolare, con l'esempio di sopra, l'output grafico dovrebbe essere simile a questo:



Esercizio 2 (6 punti)

Scrivere una funzione `seleziona(l, n)` che ha come parametri una lista l ed un numero intero n . La funzione restituisce una nuova lista ottenuta selezionando n elementi scelti casualmente da l . Lo stesso elemento non può essere selezionato più volte. Si può assumere che la funzione venga sempre invocata con n minore o uguale alla lunghezza di l . La lista l non deve essere modificata dalla funzione.

Ad esempio, alcuni valori validi come risultato di `seleziona([1, 2, 4, 8, 10], 3)` sono `[1, 2, 4]`, `[10, 2, 8]` e `[8, 2, 1]` ma non `[1, 4, 1]` (perché 1 è ripetuto più volte) o `[8, 2]` (perché ha soli due elementi).

Esercizio 3 (3 punti)

Scrivere alcuni test con il framework `pytest` per la funzione `seleziona` dell'esercizio precedente. Si considerino almeno i seguenti casi:

- un caso in cui la lista l è vuota;
- un caso in cui la lista l contiene un singolo elemento;
- un caso in cui selezionate 4 elementi da una lista di 10 elementi; in questo caso, essendo il risultato casuale, non potete controllare il suo valore esatto, ma solo che abbia le proprietà richieste (per esempio, che sia lungo 4).

Programmazione e Algoritmi 1 - Prova scritta

A.A. 2025/26 — Compito A del 15/1/2026

prof. Gianluca Amato

Esercizio 4 (5 punti)

Scrivere una funzione `list_less` che prende come parametri due liste della stessa lunghezza e le confronta elemento per elemento. La funzione restituisce `True` se tutti gli elementi della prima lista sono minori (in senso stretto) degli elementi corrispondenti della seconda lista, altrimenti restituisce `False`. Ad esempio, `list_less([1, 2, 3], [10, 20, 0])` restituisce `False` perché sebbene $1 < 10$ e $2 < 20$ non è vero che $3 < 0$.

Esercizio 5 (4 punti)

Si supponga che la cartella corrente sia `/home/giuseppe` e sia al momento vuota. Si consideri il seguente elenco di istruzioni, da eseguire una dopo l'altra tramite la shell di Linux. Si descriva brevemente cosa fa ogni singola istruzione e si disegni l'albero delle cartelle e dei file che si ottiene alla fine dell'esecuzione di tutte le istruzioni. È possibile che siano presenti nell'elenco anche istruzioni che generano errori.

```
touch gatto
mkdir animali
mv gatto animali
ls -l
cd prova
touch animali/cane
cp /home/giuseppe/animali/cane .
```

Esercizio 6 (8 punti)

Eseguire passo-passo il seguente codice Python, utilizzando l'apposito modulo:

```
1 def f(s):
2     if s == "":
3         return 0
4     elif s[0] in "aeiou":
5         return 1 + f(s[1:])
6     else:
7         return f(s[1:])
8
9 print(f("asia"))
```

Programmazione e Algoritmi 1 - Prova pratica

A.A. 2025/26 — Compito B del 15/1/2026

prof. Gianluca Amato

Esercizio 1 (8 punti)

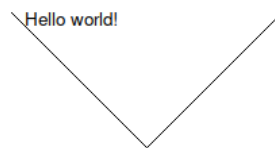
Usando la libreria `ezgraphics`, scrivere un programma che legge il file `disegno.txt` e disegna su una finestra grafica in accordo alle istruzioni ivi contenute. In particolare, il file `disegno.txt` contiene una sequenza di righe di testo. Ogni riga è una istruzione che indica che cosa disegnare, e può essere in uno di due formati:

- una lettera `l` seguita da quattro numeri interi: in questo caso bisogna disegnare una linea. I quattro numeri sono le coordinate dei punti iniziali e finali della linea, nell'ordine x_1, y_1, x_2, y_2 .
- una lettera `t` seguita da due numeri interi seguiti da una stringa qualunque: in questo caso bisogna scrivere sulla finestra grafica la stringa indicata alle coordinate specificate dai due numeri interi.

La lettera iniziale, i numeri e la stringa (nel caso di `t`) sono separati tra loro da spazi. Ad esempio, se il file `disegno.txt` contiene:

```
l 0 0 100 100
t 0 0 Hello world!
l 100 100 200 0
```

il programma dovrà disegnare qualcosa di simile a questo:



Esercizio 2 (6 punti)

Scrivere una funzione `estratto(l, n, l1)` che ha come parametri una lista l , un numero intero n ed una seconda lista l_1 . La funzione restituisce `True` se è possibile ottenere l_1 prendendo n elementi da l , senza ripetizioni.

Ad esempio, `estratto([1, 2, 4, 8, 10], 3, [1, 2, 4])` ed `estratto([1, 2, 4, 8, 10], 3, [10, 2, 8])` restituiscono il valore `True`, mentre `estratto([1, 2, 4, 8, 10], 3, [1, 4, 1])` ed `estratto([1, 2, 4, 8, 10], 3, [8, 2])` restituiscono `False` (il primo perché per ottenere `[1, 4, 1]` bisognerebbe scegliere due volte il numero 1, il secondo perché `[8, 2]` ha solo due elementi).

Esercizio 3 (3 punti)

Scrivere alcuni test con il framework `pytest` per la funzione `estratto` dell'esercizio precedente. Si considerino i seguenti casi:

- due casi in cui la lista l è vuota (uno in cui la funzione restituisce `True`, uno in cui restituisce `False`);
- due casi in cui la lista l contiene un solo elemento (uno in cui la funzione restituisce `True`, uno in cui restituisce `False`);
- tutti i casi dell'esempio qui sopra.

Programmazione e Algoritmi 1 - Prova scritta

A.A. 2025/26 — Compito B del 15/1/2026

prof. Gianluca Amato

Esercizio 4 (5 punti)

Scrivere una funzione `list_less(l, v)` che prende come parametri una lista `l` e un valore `v` e restituisce una nuova lista ottenuta da `l` selezionando i soli valori minori o uguali a `v`. Ad esempio `list_less([7, 4, 2, 10, 1, 5], 6)` restituisce `[4, 2, 1, 5]`.

Esercizio 5 (4 punti)

Si supponga che la cartella corrente sia `/home/studente` e sia al momento vuota. Si consideri il seguente elenco di istruzioni, da eseguire una dopo l'altra tramite la shell di Linux. Si descriva brevemente cosa fa ogni singola istruzione e si disegni l'albero delle cartelle e dei file che si ottiene alla fine dell'esecuzione di tutte le istruzioni. È possibile che siano presenti nell'elenco anche istruzioni che generano errori.

```
mkdir prova
touch prova/esempio
ls -l
cd prova
cp esempio ..
mv /home/studente/esempio esempio2
```

Esercizio 6 (8 punti)

Eseguire passo-passo il seguente codice Python, utilizzando l'apposito modulo:

```
1  def f(l):
2      c = 0
3      for row in l:
4          for j in range(len(row)):
5              row[j] = c
6              c += 1
7
8  l1 = [0, 0, 0]
9  l2 = [l1, l1]
10 f(l2)
11 print(l2)
```

Inoltre, descrivere la struttura della memoria, in termini di heap e stack, alla fine dell'esecuzione dell'istruzione alla riga 9.

Programmazione e Algoritmi 1 - Prova pratica

A.A. 2025/26 — Compito C del 15/1/2026

prof. Gianluca Amato

Esercizio 1 (8 punti)

Usando la libreria `ezgraphics`, scrivere un programma che disegna una spezzata (ovvero una sequenza di segmenti collegati l'uno all'altro) seguendo le istruzioni contenute nel file `disegno1.txt`. Il file in questione è composto da due righe, ognuna delle quali contiene una sequenza di numeri interi separati da spazi. Queste due righe rappresentano le coordinate di una serie di punti: in particolare, la prima riga contiene le coordinate x dei punti, mentre la seconda riga contiene le coordinate y . Il compito del vostro programma è disegnare una spezzata unendo i punti indicati nel file.

Ad esempio, supponiamo che il file `disegno1.txt` contenga quanto segue:

```
100 100 200 200
100 200 200 100
```

Il programma deve tracciare una linea dal punto di coordinate (100, 100) al punto (100, 200), poi una linea dal punto (100, 200) al punto (200, 200), infine una linea dal punto (200, 200) al punto (200, 100). In particolare, con l'esempio di sopra, l'output grafico dovrebbe essere simile a questo:



Esercizio 2 (6 punti)

Scrivere una funzione `estrai(l, n)` che ha come parametri una lista l ed un numero intero n . La funzione restituisce una nuova lista ottenuta estraendo n elementi scelti casualmente da l . Lo stesso elemento non può essere estratto più volte. Si può assumere che la funzione venga sempre invocata con n minore o uguale alla lunghezza di l . La lista l non deve essere modificata dalla funzione.

Ad esempio, alcuni valori validi come risultato di `estrai([1, 2, 4, 8, 10], 3)` sono `[1, 2, 4]`, `[10, 2, 8]` e `[8, 2, 1]` ma non `[1, 4, 1]` (perché 1 è ripetuto più volte) o `[8, 2]` (perché ha soli due elementi).

Esercizio 3 (3 punti)

Scrivere alcuni test con il framework `pytest` per la funzione `estrai` dell'esercizio precedente. Si considerino almeno i seguenti casi:

- un caso in cui la lista l è vuota;
- un caso in cui la lista l contiene un singolo elemento;
- un caso in cui estraete 3 elementi da una lista di 8 elementi; in questo caso, essendo il risultato casuale, non potete controllare il suo valore esatto, ma solo che abbia le proprietà richieste (per esempio, che sia lungo 3).

Programmazione e Algoritmi 1 - Prova scritta

A.A. 2025/26 — Compito C del 15/1/2026

prof. Gianluca Amato

Esercizio 4 (5 punti)

Scrivere una funzione `list_more` che prende come parametri due liste della stessa lunghezza e le confronta elemento per elemento. La funzione restituisce `True` se tutti gli elementi della prima lista sono maggiori o uguali degli elementi corrispondenti della seconda lista, altrimenti restituisce `False`. Ad esempio, `list_more([10, 20, 0], [10, 2, 3])` restituisce `False` perché sebbene $10 \geq 10$ e $20 \geq 2$ non è vero che $0 \geq 3$.

Esercizio 5 (5 punti)

Si supponga che la cartella corrente sia `/home/paolo` e sia al momento vuota. Si consideri il seguente elenco di istruzioni, che vengono eseguite dalla shell di Linux una dopo l'altra. Si descriva brevemente cosa fa ogni singola istruzione e si disegni l'albero delle cartelle e dei file che si ottiene alla fine dell'esecuzione di tutte le istruzioni. È possibile che siano presenti nell'elenco anche istruzioni che generano errori.

```
mkdir /home/paolo/materie
touch materie/matematica
cd materie
mv matematica ..
cd materie
cd ..
cp matematica materie/economia
```

Esercizio 6 (7 punti)

Eseguire passo-passo il seguente codice Python, utilizzando l'apposito modulo:

```
1  def f(s):
2      if s == "":
3          return 0
4      elif s[0] in "aeiou":
5          return 1 + f(s[1:])
6      else:
7          return f(s[1:])
8
9  print(f("asia"))
```

Programmazione e Algoritmi 1 - Prova pratica

A.A. 2025/26 — Compito D del 15/1/2026

prof. Gianluca Amato

Esercizio 1 (8 punti)

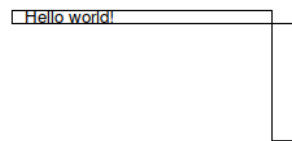
Usando la libreria `ezgraphics`, scrivere un programma che legge il file `disegno.txt` e disegna su una finestra grafica in accordo alle istruzioni ivi contenute. In dettaglio, il file `disegno.txt` contiene una sequenza di righe di testo. Ogni riga è una istruzione che indica che cosa disegnare, e può essere in uno di due formati:

- una lettera `r` seguita da quattro numeri interi: in questo caso bisogna disegnare un rettangolo. I quattro numeri sono le coordinate dei vertici opposti del rettangolo, nell'ordine x_1, y_1, x_2, y_2 .
- una lettera `t` seguita da due numeri interi seguiti da una stringa qualunque: in questo caso bisogna scrivere sulla finestra grafica la stringa indicata alle coordinate specificate dai due numeri interi.

La lettera iniziale, i numeri e la stringa (nel caso di `t`) sono separati tra loro da spazi. Ad esempio, se il file `disegno.txt` contiene:

```
r 0 0 100 100
t 10 0 Hello world!
r 100 100 200 0
```

il programma dovrà disegnare qualcosa di simile a questo:



Esercizio 2 (6 punti)

Scrivere una funzione `estratto(l, n, l1)` che ha come parametri una lista l , un numero intero n ed una seconda lista l_1 . La funzione restituisce `True` se è possibile ottenere l_1 prendendo n elementi da l , senza ripetizioni.

Ad esempio, `estratto([1, 2, 4, 8, 10], 3, [1, 2, 4])` ed `estratto([1, 2, 4, 8, 10], 3, [10, 2, 8])` restituiscono il valore `True`, mentre `estratto([1, 2, 4, 8, 10], 3, [1, 4, 1])` ed `estratto([1, 2, 4, 8, 10], 3, [8, 2])` restituiscono `False` (il primo perché per ottenere `[1, 4, 1]` bisognerebbe scegliere due volte il numero 1, il secondo perché `[8, 2]` ha solo due elementi).

Esercizio 3 (3 punti)

Scrivere alcuni test con il framework `pytest` per la funzione `estratto` dell'esercizio precedente. Si considerino i seguenti casi:

- due casi in cui la lista l è vuota (uno in cui la funzione restituisce `True`, uno in cui restituisce `False`);
- due casi in cui la lista l contiene un solo elemento (uno in cui la funzione restituisce `True`, uno in cui restituisce `False`);
- tutti i casi dell'esempio qui sopra.

Programmazione e Algoritmi 1 - Prova scritta

A.A. 2025/26 — Compito D del 15/1/2026

prof. Gianluca Amato

Esercizio 4 (5 punti)

Scrivere una funzione `list_more(l, v)` che prende come parametri una lista l e un valore v e restituisce una nuova lista ottenuta da l selezionando i soli valori strettamente maggiori di v . Ad esempio `list_more([7, 4, 2, 10, 1, 5], 6)` restituisce `[7, 10]`.

Esercizio 5 (5 punti)

Si supponga che la cartella corrente sia `/home/carlo` e sia al momento vuota. Si consideri il seguente elenco di istruzioni, che vengono eseguite dalla shell di Linux una dopo l'altra. Si descriva brevemente cosa fa ogni singola istruzione e si disegni l'albero delle cartelle e dei file che si ottiene alla fine dell'esecuzione di tutte le istruzioni. È possibile che siano presenti nell'elenco anche istruzioni che generano errori.

```
touch giuseppe
touch luisa
mkdir amici
mv luisa amici/anna
cd luisa
cp ../giuseppe michele
```

Esercizio 6 (7 punti)

Eeguire passo-passo il seguente codice Python, utilizzando l'apposito modulo:

```
1  def f(l):
2      for row in l:
3          for j in range(len(row)):
4              row[j] += 1
5
6  l1 = [1, 2, 3]
7  l2 = [l1, l1]
8  f(l2)
9  print(l2)
```

Inoltre, descrivere la struttura della memoria, in termini di heap e stack, alla fine dell'esecuzione dell'istruzione alla riga 7.