

Laboratorio di programmazione

Python

A.A. 2020-2021

Lezione 12



Esercizi per esame

Lezione 4 - Strutture dati str e dict

L4 - Esercizio 1

Prendiamo il testo seguente:

'''Nel mezzo del cammin di nostra vita
mi ritrovai per una selva oscura,
ché la diritta via era smarrita.

Ahi quanto a dir qual era è cosa dura,
esta selva selvaggia e aspra e forte,
che nel pensier rinova la paura!'''

e lavoriamoci un po' su.

1. Contate le righe (di effettivo testo) che compongono l'estratto
2. Contate le parole che compongono l'estratto
3. Scrivete al contrario il terzo verso
4. Create un dizionario che mappi ogni carattere (chiave) con la sua occorrenza nel testo (valore)
5. Create un dizionario come il precedente per le sole lettere (no caratteri speciali), ignorando maiuscole e minuscole
6. Individuate l'indice del verso che contiene la stringa "selva"
7. Individuate la posizione della stringa "selva" all'interno del verso
8. Inserite, dopo la stringa "selva" la stringa:

`"(la selva è un bosco)"`

L4 - Esercizio 2

Prendiamo la seguente rubrica (costruita con dizionari annidati):

```
{
  'Paolino Paperino': {'giorno': 9,
                      'mese': 'agosto',
                      'anno': 1934,
                      'età': 87,
                      'sesso': 'M',
                      'mail': 'paolino.paperino@disney.org'},
  'Ron Weasley': {'giorno': 1, 'mese': 'maggio', 'anno': 1980, 'età': 41, 'sesso': 'M',
                  'mail': 'ron_weasley80@hogwards.uk'},
  'Ramona Flowers': {'giorno': 18, 'mese': 'ottobre', 'anno': 2004, 'età': 42, 'sesso':
                    'F', 'mail': 'ramona.flowers@gmail.com'},
  'Madoka Ayukawa': {'giorno': 25, 'mese': 'luglio', 'anno': 1969, 'età': 52, 'sesso':
                    'F', 'mail': 'madoka_sax@asahi_net.jp'}
}
```

1. Utilizzare la rubrica costruita in precedenza per scrivere, ad ogni membro della rubrica, il seguente messaggio:

```
'''Car[o/a] [Nome],  
sei nat[o/a] il [giorno] di [mese] del [anno] e quindi a breve compirai [età]  
anni.  
Ti manderemo gli auguri a [mail]'''
```

2. A partire dalla rubrica, costruire la lista delle età, ordinata in ordine crescente
3. Invertire la lista precedentemente costruita
4. Riorganizzare la rubrica in ordine crescente di età dei personaggi

Lezione 5 - funzioni e strutture dati

L5 - Esercizio 1

- 1) Trovate 15 soluzioni per il gioco delle regine con il metodo delle permutazioni: quanto è il tempo medio?
- 2) Contate quanti tentativi fa il programma per trovare ogni soluzione
- 3) Alcune soluzioni possono essere ripetute: fate in modo che le soluzioni siano "uniche"
- 4) Se ci sono soluzioni ripetute, contate quante volte ogni soluzione è ripetuta
- 4) Generalizzate il programma per risolvere una scacchiera di qualunque dimensione $N \times N$
- 5) Trovate qual'è la scacchiera più grande di cui si riesce a trovare 1 soluzione in meno di 30s

L5 - Esercizio parte 2

7) Ogni soluzione è 'simmetrica' per rotazioni della scacchiera 8x8 di 90, 180 e 270 gradi. Trovata una soluzione, costruite le 4 simmetriche per rotazione prima di cercarne un'altra

Lezione 10 - Costruire classi

L10 - Esercizio 1

Progettare una classe `Scacchiera` per risolvere il problema delle 8 regine generalizzato che abbia le seguenti caratteristiche:

- La classe va inizializzata con la dimensione del lato. Il valore di *default* è `8` e sono ammessi valori positivi `< 15`.
- La classe ha un metodo per controllare che la dimensione della scacchiera rientri nei valori consentiti prima dell'inizializzazione.
- La classe ha un metodo per controllare che il valore di inizializzazione sia un valore tale per cui il problema ha soluzione: il problema delle N regine non è risolvibile per `N = 2` ed `N = 3`.
- La classe ha un metodo che ritorna la "soluzione" al problema delle N regine come lista di posizioni valide
- La classe deve tener traccia delle soluzioni già trovate.
- La classe ha due metodi per visualizzare le soluzioni come matrice. Un metodo **non utilizza** `numpy`, mentre il secondo **deve utilizzare** `numpy`.

La gestione degli attributi della classe è libera. Se necessario, è possibile implementare metodi di supporto.

Nel programma che userà la classe `Scacchiera`, dovrà essere utilizzata così:

- Importare la classe `Scacchiera` dal file dove è implementata.
- Dato un valore `n` per il lato della scacchiera, controllare che sia un valore valido per istanziare un oggetto `Scacchiera`.
- Creare un'istanza di `Scacchiera` per risolvere il problema di dimensione `n`.
- Chiamare più volte il metodo "soluzione" per trovare una serie di soluzioni.
- Chiamare più volte il metodo "soluzione" per trovare una serie di soluzioni univoche.
- Visualizzare la scacchiera con uno dei due metodi (numpy e non-numpy) a disposizione (o entrambi).

L10 Esercizio 2

Progettare una classe `Rubrica` per gestire nomi e indirizzi che abbia le seguenti caratteristiche:

- La classe va inizializzata passando come parametro con un dizionario con la seguente struttura:

```
{"Mario Rossi": "mario.rossi@gmail.com",  
...}
```

- La classe ha un metodo per fare l'*override* dell'inizializzazione per leggere una rubrica già pronta su un file
- La classe ha un metodo per aggiungere contatti alla rubrica
- La classe ha un metodo per salvare i contatti su file
- La classe ha un metodo per trovare tutti i contatti il cui cognome ha la stessa iniziale
- La classe ha un metodo per stampare tutto il contenuto della rubrica
- La classe ha un metodo per generare e visualizzare su schermo un messaggio di benvenuto a tutti i contatti con una specifica iniziale nel cognome, ad esempio:

```
"Ciao Mario Rossi, questo è il tuo indirizzo email: -mario.rossi@gmail.com-"
```

La gestione degli attributi della classe è libera. Se necessario, è possibile implementare metodi di supporto.

Nel programma che userà la classe `Rubrica`, dovrà essere utilizzata così:

- Importare la classe `Rubrica` dal file dove è implementata.
- Creare un'istanza di `Rubrica` che sia inizializzata leggendo i contatti da un file esterno
- Chiamare alcune volte il metodo per aggiungere contatti alla rubrica per inserire nuovi contatti
- Chiamare il metodo per salvare la rubrica su un file di testo
- Chiamare il metodo di benvenuto a tutti i contatti il cui cognome inizia per (ad esempio) `"R"`
- Chiamare il metodo che stampa tutto il contenuto della rubrica

Lezione 11 - Programmi

L11 Esercizio 1

Realizzare un programma che dato un file di testo in ingresso:

- visualizzi su schermo il conteggio delle righe di testo
- visualizzi su schermo il conteggio dei caratteri nel testo (senza distinzione tra maiuscole e minuscole)
- visualizzi la parola di lunghezza massima
- disegni un grafico che riporti la lunghezza delle parole (ascissa) e il numero di volte che una parola di quella lunghezza compare nel testo (ordinata)

La gestione della struttura del programma è libera. E' possibile implementarlo come script o come classe i cui metodi dell'oggetto sono chiamati in un *main*

L11 Esercizio 2

Realizzare un programma che data una funzione un intervallo ed un valore di approssimazione massima:

- calcoli l'integrale della funzione nell'intervallo campionando la funzione in un numero di punti di *default*
- incrementi ciclicamente il numero di punti di campionamento per migliorare la stima dell'intervallo
- interrompa la stima dell'integrale quando la differenza tra valore precedente ed attuale è minore del valore di approssimazione massima definito
- visualizzi il valore finale dell'integrale ed il numero di iterazioni eseguito per ottenerla
- prepari un grafico che visualizzi la funzione *campionata* in 3 momenti: alla prima stima *default*, a metà del numero totale di iterazioni, la curva finale corrispondente al risultato
- definita una soglia di valore (y) preparare un grafico con solo la porzione di funzione al di sopra della soglia

La gestione della struttura del programma è libera. E' possibile implementarlo come script o come classe i cui metodi dell'oggetto sono chiamate in un *main*