

Soluzioni degli esercizi

Queste soluzioni sono proposte soprattutto per favorire un'acquisizione progressiva delle conoscenze. Bisogna partire dall'assunto che esse *non* siano le uniche o le migliori. Prima di studiarle, ognuno deve cercare in autonomia le *proprie*, che potranno anche essere molto diverse da quelle proposte. Alcune delle soluzioni seguenti potrebbero essere incomplete e presentare solo alcune idee per risolvere gli aspetti più critici del problema.

Esercizi capitolo 10 - File

Resistenze da file

```
total = 0
total_inv = 0

with open("resistors.txt") as file:
    for line in file:
        val = float(line)
        if val > 0:
            total += val
            total_inv += 1 / val

if total > 0:
    print(total, 1 / total_inv)
```

 https://fondinfo.github.io/play/?exs/c10_resistors.py

Sequenza di valori

```
filename = input("filename? ")
minval, maxval = math.inf, -math.inf
with open(filename) as f:
    for line in f:
        val = float(line)
        if val < minval:
            minval = val
        if val > maxval:
            maxval = val
print(minval, maxval)
```


 https://fondinfo.github.io/play/?exs/c10_filerange.py

Fusione

```
with open("_file1.dat") as f1, open("_file2.dat") as f2:
    a = f1.readline()
    b = f2.readline()
    while a or b:
        if a and (not b or float(a) < float(b)):
            print(a.strip())
            a = f1.readline()
        else:
            print(b.strip())
            b = f2.readline()
```

 https://fondinfo.github.io/play/?exs/c10_merge.py

Cifrario monoalfabetico

 https://fondinfo.github.io/play/?exs/c10_???.py

Diagonale CSV

Trattandosi di dati numerici, possiamo riutilizzare le funzioni di lettura e scrittura CSV già viste. Resta da svolgere solo l'elaborazione dei dati sulla diagonale.

```
from random import randint

def main():
    write_csv("_data.csv", [randint(1, 12) for i in range(20)], 4, 5)

    data, cols, rows = read_csv("_data.csv")

    for i in range(min(rows, cols)):
        x, y = cols - 1 - i, rows - 1 - i
        data[y * cols + x] **= 2 # v = v ** 2


    write_csv("_data2.csv", data, cols, rows)
```

 https://fondinfo.github.io/play/?exs/c10_csvdiag.py

Primi gemelli

 https://fondinfo.github.io/play/?exs/10_???py

Film e attori

 https://fondinfo.github.io/play/?exs/c10_???py

Mappa per Pac-Man

```
PACMAN, WALL, COOKIE, POWERUP, EMPTY = "C#-+ "  
  
walls, cookies, powerups, pacman = [], [], [], None  
with open("p43_pacmanmap.txt") as file1:  
    board = [list(line.rstrip("\n")) for line in file1]  
    cols, rows = len(board[0]), len(board)  
    for y in range(rows):  
        for x in range(cols):  
            c = board[y][x]  
            if c == WALL:  
                w, h = expand_wall(board, x, y) # or, (1, 1)  
                walls.append((x*8, y*8, w*8, h*8))  
            elif c == COOKIE:  
                cookies.append((x*8+6, y*8+6))  
            elif c == POWERUP:  
                powerups.append((x*8+4, y*8+4))  
            elif c == PACMAN:  
                pacman = (x*8, y*8)  
  
with open("p32_pacmanmap.py", "w") as file2:  
    print("size =", (cols*8, rows*8), file=file2)  
    print("pacman =", pacman, file=file2)  
    print("walls =", walls, file=file2)  
    print("cookies =", cookies, file=file2)  
    print("powerups =", powerups, file=file2)
```

 https://fondinfo.github.io/play/?exs/c10_readmap.py