FIȘĂ DE LUCRU · PYTHON · CLASA A IX-A · FIȘA 13 · LECȚIA PY9-13

Fișiere text: citire, scriere și pregătirea datelor

Durată: 100 min Nivel: mate-info · Python

CE VEI ȘTI DUPĂ LECȚIE

- Prezici ce întorc read, readline, readlines și ce conține un fișier după write, print(file=...) și modurile "w"/"a".
- Repari erorile tipice: mod greșit, conversie uitată, fișier gol, write cu numere.
- Scrii programe care citesc, prelucrează din mers și scriu fișiere, testate și pe cazuri limită.

Ce trebuie să știi (o pagină)

Ce vrei	Cum scrii	De reținut
deschizi pentru citire	`with open("a.txt", encoding="utf-8") as f:`	fișier inexistent → `FileNotFoundError`
deschizi pentru scriere	`with open("b.txt", "w", encoding="utf-8") as g:`	**șterge** conținutul vechi
adaugi la sfârșit	`open("b.txt", "a", …)`	păstrează conținutul
tot restul fișierului	`f.read()`	un singur `str`; la sfârșit `''`
un rând	`f.readline()`	cu `\n` la final; la sfârșit `''`
toate rândurile	`f.readlines()`	listă de `str`, fiecare cu `\n`
rând cu rând	`for linie in f:`	cel mai des folosit
toate numerele, oricum așezate	`[int(x) for x in f.read().split()]`	`split()` ignoră spațiile și rândurile goale
câteva valori de pe un rând	`a, b = map(int, f.readline().split())`	
scrii text	`g.write("…\n")`	doar `str`, fără `\n` automat
scrii ca pe ecran	`print(x, y, file=g)`	convertește și trece la rând nou
fișier lipsă	`try: … except FileNotFoundError: …`	
fișier gol	verifici `len(valori) == 0` înainte de medie	altfel `ZeroDivisionError`

Nivelul 1 · Prezice

★ citește și prezice

În enunțuri, ↵ marchează sfârșitul unui rând (caracterul `\n`).

P1. Fișierul `a.txt` conține `7↵3 4↵↵final↵`. Ce afișează?

```
with open("a.txt") as f:
    print(repr(f.readline()), f.readline().split(), repr(f.readline()), repr(f.read()))
```

P2. Pentru același fișier, după `r = f.readlines()`: cât este `len(r)`, ce dă `repr(r[2].strip())` și ce dă `repr(r[-1])`?

P3. Fișierul `b.txt` conține `abc↵de↵`. Ce afișează?

```
with open("b.txt") as f:
    x = f.read()
    y = f.read()
print(len(x), len(y))
```

P4. Ce conține `c.txt` (scrie-l cu `↵`)?

```
with open("c.txt", "w") as g:
    g.write("1")
    g.write("2")
    print(3, file=g)
    g.write("4")
```

P5. Ce conține `d.txt` la final?

```
for mod, text in [("w", "A\n"), ("a", "B\n"), ("w", "C\n"), ("a", "D\n")]:
    with open("d.txt", mod) as g:
        g.write(text)
```

P6. Fișierul `e.txt` conține `x↵y↵`. Ce apare pe ecran cu `for linie in f: print(linie)`? Dar cu `print(linie, end="")`?

P7. Fișierul `n.txt` conține `1↵2↵3↵`. Ce afișează?

```
s = ""
with open("n.txt") as f:
    for linie in f:
        s += linie.strip()
print(s, int(s) + 1)
```

P8. Fișierul `note.csv` conține `Ana,9,10↵Ion,7,8↵`. Ce afișează?

```
with open("note.csv") as f:
    for linie in f:
        p = linie.strip().split(",")
        print(p[0], (int(p[1]) + int(p[2])) / 2)
```

Nivelul 2 · Găsește și repară

★★ găsește și repară

P9. `date.txt` conținea `5 6 7`. După rularea programului apare `io.UnsupportedOperation: not readable`, iar fișierul este acum **gol**. Explică ambele efecte și repară.

```
with open("date.txt", "w") as f:
    x = f.read()
```

P10. `n.txt` conține `4↵5↵6↵`. Programul dă `TypeError: unsupported operand type(s) for +=: 'int' and 'str'`. Repară-l. Trebuie să tai și `\n`-ul înainte de conversie?

```
suma = 0
with open("n.txt") as f:
    for linie in f:
        suma += linie
print(suma)
```

P11. Programul calculează media numerelor din `temp.txt`. Pe un fișier gol se oprește cu `ZeroDivisionError`. Adaugă verificarea care lipsește și afișează `fișier gol`.

```
with open("temp.txt") as f:
    v = [float(x) for x in f.read().split()]
print(sum(v) / len(v))
```

P12. *Completează.* `numar.txt` conține un număr natural nenul `n`. Programul scrie în `rezultate.txt`, pe trei rânduri: `n²`, rădăcina pătrată a lui `n` cu două zecimale și divizorii lui `n`, separați prin spații. Pentru `12` fișierul trebuie să fie `144↵3.46↵1 2 3 4 6 12 ↵`.

```
import math
with open("numar.txt", encoding="utf-8") as f:
    n = ________(f.________())
with open("rezultate.txt", ________, encoding="utf-8") as g:
    print(n * n, ________)
    g.write(f"{math.sqrt(n):.2f}________")
    for d in range(1, n + 1):
        if n % d == 0:
            g.write(________ + " ")
    g.write("\n")
```

P13. *Rânduri amestecate.* Programul copiază `intrare.txt` în `iesire.txt`, cu litere mari. Un rând este în plus: care și ce ar strica?

```
A)  g.write(linie.upper())
B)  for linie in f:
C)  f = open("intrare.txt", encoding="utf-8")
D)  g = open("iesire.txt", "r", encoding="utf-8")
E)  f.close()
F)  g = open("iesire.txt", "w", encoding="utf-8")
G)  g.close()
```

Pentru `Ana are mere.↵Ion are pere.↵` ce conține `iesire.txt`? Rescrie apoi programul cu `with`.

P14. Programul trebuie să scrie numerele de la 1 la 5, câte unul pe rând. Găsește **cele două** probleme și dă două reparații diferite.

```
with open("p14.txt", "w") as g:
    for i in range(1, 6):
        g.write(i)
```

Nivelul 3 · Scrie cod

★★ scrie cod

Creează fișierele de intrare din Python (cu `"w"`), cu exact conținutul din tabel, și verifică **toate** testele.

P15. Stația meteo. `temp.txt` conține temperaturi reale (pot fi negative), câte una pe rând (pot exista rânduri goale și spații). Afișează minimul, maximul și media cu o zecimală, sau `fișier gol`, sau `fișierul lipsește`.

`temp.txt`	ecran
`3.5↵-2↵7↵0.5↵`	`-2.0 7.0 2.2`
`-4↵-1.5↵-3↵`	`-4.0 -1.5 -2.8`
`12↵`	`12.0 12.0 12.0`
`8↵↵ 9 ↵`	`8.0 9.0 8.5`
(gol)	`fișier gol`
(fișierul nu există)	`fișierul lipsește`

P16. Catalogul. Fiecare rând din `catalog.txt` are un nume și una sau mai multe note. Scrie în `medii.txt`, pentru fiecare elev, `Nume media` (două zecimale). Pe ecran afișează elevul cu media cea mai mare (primul, la egalitate).

`catalog.txt`	`medii.txt`	ecran
`Ana 9 10 8↵Bogdan 7 6↵Cristina 10 9 10 9↵Dan 5↵`	`Ana 9.00↵Bogdan 6.50↵Cristina 9.50↵Dan 5.00↵`	`Cristina`
`Ion 8 9↵Mara 9 8↵`	`Ion 8.50↵Mara 8.50↵`	`Ion`

P17. Ca la Bac. `bac.txt` conține numere naturale, câte mai multe pe rând. Afișează câte numere pare sunt și cel mai mare număr par, sau `nu exista`. Prelucrează din mers, rând cu rând, fără să păstrezi numerele.

`bac.txt`	ecran
`3 8 5↵12 7↵4↵`	`3 12`
`1 3↵5↵`	`nu exista`
`0↵`	`1 0`
`10 10↵↵2↵`	`3 10`

P18. Cel mai lung vers. Afișează numărul rândului cel mai lung din `poezie.txt` (rândurile se numără de la 1; primul, la egalitate) și lungimea lui, fără `\n`.

`poezie.txt`	ecran
`Somnoroase păsărele↵Pe la cuiburi se adună,↵Se ascund în rămurele -↵Noapte bună!↵`	`2 23`

poezie.txt	ecran
`ab↵cd↵`	`1 2`
`x` (fără `↵` la final)	`1 1`

P19. Cheltuieli. `cheltuieli.csv` are pe primul rând capul de tabel `data,categorie,suma`, apoi câte o cheltuială pe rând. Citește de la tastatură o categorie și afișează, cu două zecimale, totalul ei și totalul general.

Fișierul: `data,categorie,suma↵2026-10-01,mancare,45.50↵2026-10-01,transport,4↵2026-10-02,mancare,32↵2026-10-03,carti,59.90↵`

categorie citită	ecran
`mancare`	`77.50 141.40`
`transport`	`4.00 141.40`
`haine`	`0.00 141.40`

P20. Zarul în fișier. Citește un `seed`. Cu `random.Random(seed)` scrie în `zaruri.txt` 1000 de aruncări (`randint(1, 6)`), câte una pe rând. Apoi **recitește** fișierul și afișează câte valori are, de câte ori a ieșit 6 și media cu două zecimale.

seed	ecran
`1`	`1000 163 3.53`
`2026`	`1000 177 3.61`

Nivelul 4 · Pune mintea la contribuție

★★★ pune mintea la contribuție

P21. Cazuri limită. O soluție „naivă" pentru P15 este `t = [float(x) for x in open("temp.txt").read().split()]` urmată de `print(min(t), max(t), sum(t) / len(t))`. Pentru fiecare caz, spune dacă soluția naivă merge sau ce eroare dă: fișier inexistent; fișier gol; o singură valoare; doar valori negative; rânduri goale între valori; `2,5` în loc de `2.5`.

P22. Fișier sau tastatură? Scrie un program care generează un fișier cu 1 000 000 de numere aleatoare (câte unul pe rând) și măsoară, cu `time.perf_counter()`, cât durează să le aduni citindu-le (a) cu `f.read().split()` și (b) rând cu rând. Compară cu timpul de care ar avea nevoie un om să tasteze 1 000 000 de numere. Care variantă folosește mai puțină memorie?

P23. Fișierul ascuns. Programul de mai jos a afișat `2 3`. Scrie un conținut posibil, cu 5 rânduri, pentru `x.txt`. Ce s-ar fi afișat dacă unul dintre rânduri conținea doar spații?

```
goale = pline = 0
with open("x.txt") as f:
    for linie in f:
        if linie.strip() == "":
            goale += 1
        else:
            pline += 1
print(goale, pline)
```

P24. De ce `''` și nu `strip()`? Rescrie suma numerelor din fișier (câte un număr pe rând, pot exista rânduri goale) folosind `while True`, `readline()` și `break`. Un coleg oprește bucla cu `if linie.strip() == "": break`. Pe fișierul `1↵2↵↵3↵`, ce afișează varianta ta și ce afișează varianta colegului? De ce?

P25. Mini-proiect: jurnalul de alergare. Scrie funcțiile `adauga(data, km)`, care adaugă un rând `data km` în `alergare.txt` (fără să șteargă ce era), și `raport()`, care întoarce totalul și ziua cu cea mai lungă alergare. Programul principal citește o dată și un număr de km, apelează `adauga`, apoi afișează `raport()`. Rulează-l de trei ori, cu `2026-10-26 5`, `2026-10-27 7.5`, `2026-10-29 3.2`. Ce afișează la a treia rulare? De ce aici este nevoie de modul `"a"` și nu de `"w"`?

P26. Feedback pentru un coleg. Un coleg a rezolvat P18 așa:

```
f = open("poezie.txt")
mx = 0
nr = 0
i = 0
for linie in f:
    i += 1
    if len(linie) >= mx:
        mx = len(linie)
        nr = i
print(nr, mx)
```

a. Completează grila (0 = nu, 1 = parțial, 2 = da): corect pe cele trei teste din enunț; corect la egalitate; tratează fișierul fără `↵` final; închide fișierul; nume de variabile clare. b) Scrie pentru coleg două puncte tari și două sugestii, fiecare cu un exemplu concret. c) **În clasă:** rezolvă P18, schimbă foaia cu un coleg și completează aceeași grilă pentru soluția lui. Discutați punctajele diferite.

Probleme de lucru în clasă (pbinfo)

Le rezolvăm împreună în ședință, în ordinea de mai jos.

- [ ] **#1 sum** — ușoară
  https://www.pbinfo.ro/probleme/1/sum
- [ ] **#57 n-suma** — ușoară
  https://www.pbinfo.ro/probleme/57/n-suma
- [ ] **#7 maxim2** — ușoară
  https://www.pbinfo.ro/probleme/7/maxim2

Temă pentru acasă (pbinfo)

Trimite soluțiile pe pbinfo și notează punctajul obținut. Bifează ce ai terminat.

- [ ] #8 **maxim3** — ușoară
  https://www.pbinfo.ro/probleme/8/maxim3
- [ ] #2269 **maximpar** — ușoară
  https://www.pbinfo.ro/probleme/2269/maximpar
- [ ] #3391 **sirmaxim** — ușoară
  https://www.pbinfo.ro/probleme/3391/sirmaxim
