

**Programarea Algoritmilor**  
**– LABORATOR NR. 2 –**

<https://docs.python.org/3/library/stdtypes.html#text-sequence-type-str>

1. Se citește un cuvânt. Să se ștergă din cuvânt toate aparițiile primei litere. Se va afișa un mesaj de forma: După ștergerea literei 'X' șirul obținut este "S" de lungime L folosind diferite tipuri de formatare (cu parametri poziționali și f-stringuri)
2. Să se verifice (folosind întâi metoda find, apoi index) dacă un șir de caractere  $t$  apare ca subșir într-un șir  $s$ , iar în caz afirmativ să se afișeze toate pozițiile la care începe  $t$  în  $s$ . De exemplu, șirul  $t = "abc"$  apare ca subșir în șirul  $s = "abccabcababcc"$  începând cu pozițiile 0, 4 și 9.
3. Se citește un cuvânt  $s$  de cel mult 10 de caractere. Sa se afișeze (folosind  $s[i:j]$ ) pe câte o linie cuvintele obținute succesiv din  $s$  tăind prima și ultima literă (afișate centrat pe 10 de caractere):

```
algoritm
lgorit
gori
or
```

4. Într-o propoziție a fost efectuată, posibil de mai multe ori, aceeași greșeală de ortografie.
  - a) Scrieți un program care citește propoziția, șirul greșit și șirul corect, după care afișează propoziția corectă. De exemplu, în propoziția "Problemele cu șiruri de caracter nu sunt ggerle!" greșeală constă în faptul că în loc de șirul "re" a fost scris șirul "ger".
  - b) Modificați programul astfel încât să corecteze maxim 2 astfel de greșeli, iar dacă sunt mai multe să afișeze mesajul: "textul conține prea multe greseli, doar 2 au fost corectate"
5. Scrieți un program care să înlocuiască într-o propoziție toate aparițiile unui cuvânt  $s$  cu un cuvânt  $t$  (!cuvânt, nu subșir). Cuvintele sunt separate prin unul sau mai multe spații.
6. Se citește un text codificat după regula: În fata fiecărui caracter este scris un număr de cel mult 2 cifre care reprezintă numărul de apariții consecutive ale acestui.
  - a) Scrieți un program care decodifica textul. De exemplu textul 1G10o4l se va decodifica GoooooooooIIII
  - b) Scrieți un program care, dat un text, îl codifică după regula de la a)
7. Jurnalul electronic al Anei conține, în fiecare zi, câte o frază cu informații despre cheltuielile pe care ea le-a efectuat în ziua respectivă. Scrieți un program care să citească o frază de acest tip din jurnalul Anei și apoi să afișeze suma totală cheltuită de ea în ziua respectivă. De exemplu, pentru fraza "Astăzi am cumparat paine de 5 RON, pe lapte am dat 10 RON, iar de 15 RON am cumparat niste cascaval. De asemenea, mi-am cumparat

si niste papuci cu 50 RON!”), programul trebuie să afișeze suma totală de 80 RON. Fraza se consideră corectă, adică toate numerele care apar în ea sunt numere naturale reprezentând sume cheltuite de Ana în ziua respectivă!

#### 8. Numele Pre-Nume

Scrieți un program care citește un șir de caractere și decide dacă acesta este un nume complet corect al unei persoane. Se consideră că un nume complet este corect dacă respectă următoarele proprietăți:

- persoana poate avea cel mult doua prenume, iar dacă sunt două atunci sunt despărțite printr-o cratimă ('-'). La fel și în cazul numelui de familie
- numele de familie sau prenume conține doar litere și cel mult o cratimă.
- orice nume de familie sau prenume este format din cel puțin 3 litere.
- orice nume sau prenume începe cu literă mare.

Exemplu de nume complete corecte: Ionescu-Cherea Mihai-Adrian, Popescu Elena-Maria, Vlad Matei și de nume incorecte: Ionescu - Cherea Mihai, Vlad Matei Alexandru

#### 9. Cifrul lui Cezar

a) Se citește un text și un număr natural  $k$ . Să se afișeze textul cifrat cu cifrul lui Cezar, prin care fiecare literă (!doar literele) este înlocuită cu litera aflată peste  $k$  poziții la dreapta în alfabet în mod circular (valoarea  $k$  reprezintă cheia secretă comună pe care trebuie să o cunoască atât expeditorul, cât și destinatarul mesajului criptat).

b) Se citește un număr natural  $k$  și text criptat cu cifrul lui Cezar cu cheia  $k$ . Să se afișeze textul decriptat

#### 10. Se citesc doua cuvinte formate doar din litere mici. Să se verifice dacă sunt anagrame.

11. a) Se citește de la tastatură un text. Se cere să se “traducă” în limba păsărească textul dat astfel: după fiecare vocală se adaugă litera p și încă o dată acea vocală (după a, e, i, o, u se adaugă respectiv pa, pe, pi, po, pu). Exemplu: “Ana are mere.” devine “Apanapa aparepe meperepe.” Fiind dat un astfel de text în limba păsărească, se poate obține textul original? Dacă da, scrieți un program care primind un text în limba păsărească construiește în memorie și afișează textul inițial.

b) Se citește de la tastatură un text în care cuvintele sunt despărțite în silabe cu ajutorul cratimelor. Se cere să se “traducă” textul dat în limba păsărească astfel: după fiecare silabă se adaugă litera p și se repetă ultima literă din acea silabă. Afișați traducerea și cu cratime, dar și fără.

Exemplu: “a-na a-re mul-te me-re ro-sii si de-li-cioa-se.” devine

“apa-napa apa-repe mulpl-tepe mepe-repe ro-po-siipi sipi depe-lipi-cioapa-sepe.” și

“apanapa aparepe mulpltepe meperepe roposiipi sipi depelipicioapasepe.”

Fiind dat un astfel de text în limba păsărească (cel care conține și cratime), se poate obține textul original? Dacă da, scrieți un program care decodifică un astfel de text.