

PRAKTIKUM IZ PROGRAMIRANJA 1

- domaći zadatak broj 4 -

Sastaviti program na programskom jeziku Python, kojim se vrši određena vrsta obrade nad tekstualnim podacima (stringovima). Glavni program treba da:

- učitava jednu ili više linija teksta sa kojim će raditi;
- poziva odgovarajuću funkciju koja izvrši zahtevanu obradu nad zadatim tekstom;
- ispisuje sve dobijene rezultate;

Zavisno od rednog broja problema, svaki student treba da sastavi jedan od programa koji su dati u prilogu ovog dokumenta. Voditi računa o dekompoziciji programa na potprograme prema gore navedenoj raspodeli i zahtevima pojedinačnih zadataka. Po potrebi, dozvoljeno je uvoditi i dodatne potprograme u rešenje. **Potprogrami ne smeju pristupati promenljivama glavnog programa direktno, već samo putem svojih argumenata i povratne vrednosti.**

Radi boljeg testiranja programa, odabrati nekoliko dodatnih skupova podataka sa kojima će program biti testiran. Svaki primer treba da sadrži ulazne podatke i očekivani izlaz za te podatke.

Napomene:

- Rok za predaju četvrtog domaćeg zadatka je **ponedeljak, 25.12.2023.** putem kursa predmeta na **Moodle** platformi za elektronsko učenje. Tačan termin za predaju će biti naknadno definisan za sve studente. Termin će biti ograničenog vremenskog trajanja.
- Domaći zadaci će biti testirani i ocenjivani korišćenjem javnih i tajnih testova.
- Studentima će nekoliko dana pre roka za predaju biti dostupno okruženje za testiranje rešenja domaćeg zadatka na Moodle platformi za elektronsko učenje korišćenjem javnih testova.
- Prilikom predaje domaćeg zadatka studenti će rešavati i kratak test znanja u vezi rešenja domaćeg zadatka i relevantnog gradiva iz oblasti programiranja na programskom jeziku Python koje obuhvata temu domaćeg zadatka.
- Domaći zadaci se rešavaju **samostalno**. Predmetni nastavnici zadržavaju pravo da nakon predaje domaćih zadataka izvrše proveru sličnosti i preuzmu odgovarajuće disciplinske mere.
- Svi drugi detalji oko predaje, ocenjivanja i odbrane domaćeg zadatka će biti blagovremeno objavljeni.
- Formula za redni broj problema **i** koji treba rešavati je sledeća (R – redni broj indeksa, G – poslednje dve cifre godine upisa):

$$i = (R + G) \bmod 4$$

- Kao rešenje domaćeg zadatka potrebno je predati sadržaj sledeće datoteke:

- `dz4.py`, koja sadrži izvorni tekst programa na programskom jeziku Python;

0. Napisati program koji sve reči u nekom programskom kodu koje se nalaze u camelCase formatu (više reči spojenih u jednu reč tako što svaka reč počinje velikim slovom) prebacuje u snake_case format (više reči spojenih u jednu tako da su originalne reči razdvojene znakom `_`). Ukoliko se u zadatom programu nađe znak `;`, korektno prekinuti izvršavanje programa.

Program treba da:

- 1) Učita jednu liniju teksta sa standardnog ulaza.
- 2) Obradi liniju teksta u skladu sa navedenim zahtevom i rezultat ispiše na standardni izlaz.
- 3) Ponavlja obradu sve dok se ne unese prazna linija.

Primeri

ULAZ	IZLAZ
<code>def addTwoNumbers(firstNum, secNum)</code>	<code>def add_two_numbers(first_num, sec_num)</code>
<code>numbersList = [1, 2, 3] secondList = [4, 5, 6] totalSum = sum(numbersList) + sum(secondList)</code>	<code>numbers_list = [1, 2, 3] second_list = [4, 5, 6] total_sum = sum(numbers_list) + sum(second_list)</code>
<code>a = 1 b = 2;</code>	<code>a = 1</code>

1. Napisati program koji iz zadatog teksta koji predstavlja programski kod izbacuje jednolinijske komentare. Komentar se sastoji iz znaka `#` i svih ostalih karaktera do kraja reda. Ukoliko je neka linija sadržala samo komentar potrebno je izbaciti celu liniju. Ukoliko se u zadatom programu nađe znak `;`, korektno prekinuti izvršavanje programa.

Program treba da:

- 1) Učita jednu liniju teksta sa standardnog ulaza.
- 2) Obradi liniju teksta u skladu sa navedenim zahtevom i rezultat ispiše na standardni izlaz.
- 3) Ponavlja obradu sve dok se ne unese prazna linija.

Primeri

ULAZ	IZLAZ
<code># this is a function def add_two_numbers(first_num, sec_num) #same line as function</code>	<code>def add_two_numbers(first_num, sec_num)</code>
<code>numbers_list = [1, 2, 3] # comment second_list = [4, 5, 6] # comment total_sum = sum(numbers_list) + sum(second_list) # comment</code>	<code>numbers_list = [1, 2, 3] second_list = [4, 5, 6] total_sum = sum(numbers_list) + sum(second_list)</code>
<code>a = 1 b = 2; # comment</code>	<code>a = 1</code>

2. Napisati program koji sve reči u nekom programskom kodu koje se nalaze u snake_case formatu (više reči spojenih u jednu tako da su originalne reči razdvojene znakom `_`) prebacuje u camelCase format (više reči spojenih u jednu reč tako što svaka reč počinje velikim slovom). Ukoliko se u zadatom programu nađe znak `;`, korektno prekinuti izvršavanje programa.

Program treba da:

- 1) Učita jednu liniju teksta sa standardnog ulaza.
- 2) Obradi liniju teksta u skladu sa navedenim zahtevom i rezultat ispiše na standardni izlaz.
- 3) Ponavlja obradu sve dok se ne unese prazna linija.

Primeri

ULAZ	IZLAZ
<code>def add_two_numbers(first_num, sec_num)</code>	<code>def addTwoNumbers(firstNum, secNum)</code>
<code>numbers_list = [1, 2, 3]</code> <code>second_list = [4, 5, 6]</code> <code>total_sum = sum(numbers_list) + sum(second_list)</code>	<code>numbersList = [1, 2, 3]</code> <code>secondList = [4, 5, 6]</code> <code>totalSum = sum(numbersList) + sum(secondList)</code>
<code>a = 1</code> <code>b = 2;</code>	<code>a = 1</code>

3. Napisati program koji u zadatom programu sve pozive `sort` i `reverse` metoda menja pozivima `sorted` i `reversed` funkcija. Ukoliko se u zadatom programu nađe znak `;`, korektno prekinuti izvršavanje programa.

Program treba da:

- 1) Učita jednu liniju teksta sa standardnog ulaza.
- 2) Obradi liniju teksta u skladu sa navedenim zahtevom i rezultat ispiše na standardni izlaz.
- 3) Ponavlja obradu sve dok se ne unese prazna linija.

Primeri

ULAZ	IZLAZ
<code>l = [1, 2, 3]</code> <code>l.sort()</code>	<code>l = [1, 2, 3]</code> <code>sorted(l)</code>
<code>l1 = [1, 2, 3]</code> <code>l1.reverse()</code> <code>l1.sort()</code> <code>l1.reverse()</code>	<code>l1 = [1, 2, 3]</code> <code>reversed(l1)</code> <code>sorted(l1)</code> <code>reversed(l1)</code>
<code>a = 1</code> <code>l = [1, 2, 3];</code>	<code>a = 1</code>