

1. Što će biti ispisano na ekranu nakon izvršavanja sljedećeg programa? Koje vrijednosti će poprimiti varijabla i u programu?

- ☒ A 0,1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18,19
- ☐ B 1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18,19,20

```
for i in range(20):  
    print(i, end = ' ')
```

2. Što će biti ispisano na ekranu nakon izvršavanja sljedećeg programa? Koje vrijednosti će poprimiti varijabla i u programu?

- ☐ A 2,4,6,8,10,12,14,16,18
- ☐ B 0,2,4,6,8,10,12,14,16,18,20
- ☐ C 0,2,4,6,8,10,12,14,16,18
- ☐ D 2,4,6,8,10,12,14,16,18,20

```
for i in range(0, 20, 2):  
    print(i, end = ' ')
```

3. Što treba pisati na crti da bi program ispisao:

3
11
19

for num in _____:
 print(num)

- ☐ A range(3, 8, 20)
- ☐ B range(3, 19, 8)
- ☐ C range(3, 20, 8)
- ☐ D range(3, 23, 8)

4. Kolika je vrijednost varijabli a i b na kraju izvršavanja programa?

- ☐ A a=3, b=5
- ☐ B a=3, b=8
- ☐ C a=14, b=8
- ☐ D a=14, b=5

```
a=3
b=8
if int(b/a)<3*a & b:
    a=14
else:
    b = a + 2
    print(a,b)
```

- Što će ispisati sljedeći program?

```
a = "plijeniti"  
if a[3:6] != "lij":  
    print(2+2**3+1)  
else:  
    print(40)
```

- Što će ispisati sljedeći program?

```
a=1
b=2
for abc in range(4,0,-1):
    a=b+3
    b=a+3+abc
    d=abc
print(a,b,abc,d)
```

- Napišite redom sve vrijednosti koje poprimaju sve varijable tijekom izvođenja sljedećeg pseudokoda. Što će biti ispisano?

```
a= 1+2^3+1
b= a MOD 3 + 1
c= cijeli dio od(a/4*b)
a= b+c
ako je b<apsolutna vrijednost od(a-c) onda
    b=b+a*b
inače
    b=b-a*b
c=a+b+c
ispis (a,b,c)
```

- Kako se mijenjaju vrijednosti varijabli sljedećeg programa:

```
x = 0
x += 1
print(x)
x += 1
print(x)
x **= 10
print(x)
x = 100
y = 5
x *= y + 1
print(x)
```


- Kako se mijenjaju vrijednosti varijabli i što će ispisati sljedeći program:

```
m = int(input("Prvi broj: "))
n = int(input("Drugi broj: "))
a, b = m, n
while m * n != 0:
    if m > n:
        m %= n
    else:
        n %= m

print('nzd brojeva {0} i {1} je {2}'.format(a, b, m + n))
```

- Kako se mijenjaju vrijednosti varijabli i što će ispisati sljedeći program:

```
k = int(input("Unesi broj: "))
for n in range(1, k + 1):
    a, s, p = n, 0, 0
    while n > 0:
        n //= 10
        p += 1
    n = a
    while n > 0:
        j = n % 10
        n //= 10
        s += j ** p
    if a == s:
        print(a, end=' ')
```

- Što je rezultat sljedećih naredbi?

```
s = 'ABCDEFGH'  
print(s)  
print(s[0])  
print(s[1])  
print(s[8])
```

- Što je rezultat sljedećih naredbi?

```
s = 'ABCDEFGH'  
print(s[-1])  
print(s[-2])  
print(s[-8])  
print(s[-9])
```

- Pronađi grešku u kodu:

```
a = input("Unesi broj:")

if a % 2 == 0:
    print("Broj {0} je paran.".format(a))
else:
    print("Broj {0} je neparan.".format(a))
```

- Pronađi grešku u kodu:

```
def rekurzija(n):  
    print(n)  
    rekurzija(n+1)  
  
rekurzija(1)
```

- Napiši program koji unosi troznamenkasti broj n . Obaviti provjeru unosa.
- Ispisati sve troznamenkaste brojeve do n kojima je prva znamenka manja od druge, a druga manja od treće.

- Napiši program koji će unositi string i jedan znak. Program treba ispisati sve stringove koji se mogu dobiti postavljanjem unesenog znaka na bilo koje mjesto u unesenom stringu.
- Primjer:

Unos	Ispis
ZNAK T	TZNAK ZTNAK ZNTAK ZNATK ZNAKT

- Za broj ćemo reći da je savršen ako je jednak zbroju svojih djelitelja.
- Napiši funkciju koja će vraćati True ako je broj savršen, a inače će vraćati False.

- Napisati pseudokod ili nacrtati dijagram toka.

U davnoj prošlosti kada nisu postojali moćni operacijski sustavi poput "Windows 98", računala nisu mogla prikazivati sva naša slova. Na tipkovnici se nigdje nije mogla naći oznaka NJ ili pak Ž, pa su se naši programeri morali dogovoriti kako će zapisivati naša slova na računalu.

Odlučili su svako naše slovo koje se ne pojavljuje u engleskoj abecedi prikazati kao dva ili tri uzastopna znaka prema sljedećoj tablici:

Potrebno je za riječ zapisanu u ovakvom obliku odrediti od koliko se ona zapravo slova sastoji. Na primjer, riječ zapisana kao LJES=NJAK se sastoji od 6 slova: lj, e, š, nj, a, k.

Slovo	Znakovi
LJ	LJ
NJ	NJ
Č	C-
Ć	C=
Š	S=
Ž	Z=
Đ	D-
DŽ	DZ=

Ulazni podaci:

- niz znakova sastavljen od velikih slova A . . Z i znakova - i = koji će se sastojati od najviše 30 znakova

Izlazni podaci:

- broj od koliko se slova sastoji riječ koju predstavlja taj niz znakova.

Primjeri:

Ulaz	Izlaz
LJES=NJAK	6
DZ=AK	3

- Napisati pseudokod ili nacrtati dijagram toka.

Ivičina mlađa sestra Marica učenica je drugog razreda osnovne škole. Kad ne jede napolitanke, Marica vježba matematiku, jer bi željela biti dobar matematičar kao i njen stariji brat. Savladala je tablicu množenja, pa sada zdušno vježba računanje. Naš zadatak je napraviti program pomoću kojega će Marica moći lagano provjeravati svoja rješenja jednostavnih računskih zadataka. Program treba izračunati vrijednost *aritmetičkog izraza*, poput jednostavnog kalkulatora (*digitrona*).

Aritmetički izraz sastoji se od najmanje jednog i najviše 3 cijela broja ($0 \leq \text{broj} < 100$), te nekih od znakova osnovnih aritmetičkih operacija: +, -, * i /.

+ zbrajanje, - oduzimanje, * množenje, / dijeljenje.

Znak operacije u izrazu:

- uvijek dolazi između dva broja,
- ne može biti prvi ni zadnji član izraza,
- dva znaka operacija ne mogu se pojaviti uzastopno jedan iza drugoga.

Izraz će uvijek biti zadan matematički ispravno. Rezultat dijeljenja uvijek će biti cjelobrojan i ne treba provjeravati dijeljenje s 0.

Ulazni podaci: aritmetički izraz

Izlazni podaci: vrijednost aritmetičkog izraza

Primjeri:

RB.	ULAZ	IZLAZ
1.	2+7*3	23
2.	1-10	-9
3.	5*70/10	35

- Napisati pseudokod ili nacrtati dijagram toka.

Ivica slavi rođendan i pozvao je sve prijatelje. Svi znaju da Ivica najviše voli čokoladu, te su mu je i darivali. Čokoladu su donosili u pločicama, no one nisu uvijek bile cijele.

Ivica je zapisivao koliko je tko donio čokolade, izrazivši razlomkom količinu čokolade u pločicama ($1/1 - 1$ pločica, $1/2$ -pola pločice, $4/3 - 1$ cijela i trećina pločice, itd.).

Vaš je zadatak pomoći Ivici da utvrdi koliko je čokolade ukupno dobio i koji mu je prijatelj donio više čokolade.

ULAZ:

– niz od K ($1 < K < 5$) *razlomaka* M/N međusobno odvojenih znakom + (plus).

Razlomak čine dva prirodna broja M (brojnik) i N (nazivnik) između kojih se nalazi znak / (kosa crta) i $0 < M, N < 100$.

IZLAZ:

– zbroj svih razlomaka iz ulaznog niza, prikazan kao razlomak skraćen do kraja;

– ulazni niz razlomaka sortiran (složen) u rastućem redoslijedu (od najmanjeg prema najvećem). Razlomci moraju biti ispisani u nizu (retku), međusobno odvojeni znakom + (kao i na ulazu).

Napomena: Ako su dva razlomka jednaka, prije dolazi onaj s manjim brojnikom (npr. $1/2$ dođe prije $2/4$).

RB	Ulaz	Izlaz
1.	$1/2 + 3/4$	$5/4$ $1/2 + 3/4$
2.	$10/2 + 2/6 + 5/3$	$7/1$ $2/6 + 5/3 + 10/2$
3.	$1/3 + 8/12 + 2/3$	$5/3$ $1/3 + 2/3 + 8/12$

• Napiši program u Pythonu:

Marica trenutno muči muku s matematikom. Budući da i nema baš prevelikog interesa naučiti matematiku, odlučila je svom mlađem bratu Ivici, strasnom sakupljaču sličica Pokemona, za svaki riješeni zadatak iz domaće zadaće kupiti paketić sličica. Sve bi savršeno funkcioniralo da Marica na satu nije jako nepažljiva i sve joj je važnije od matematike. Dok učiteljica piše zadatke za domaću zadaću na ploču, ona nerijetko piše poruke na svom novom mobitelu i stoga krivo prepíše zadatak. Osobito česte pogreške čini prilikom prepisivanja zagrada. Iz nekih nepoznatih razloga *lijeve* zagrade uvijek prepíše dobro, a griješi samo u prepisivanju *desnih* zagrada. *Lijeve* zagrade su: {, [, (a *desne* su: },],)

Poznato je da svaka *lijeva* zagrada ima odgovarajuću *desnu* zagradu koja ju na neki način zatvara. Maričine pogreške su takve da za odgovarajuću *lijevu* zagradu prepíše pogrešnu *desnu* zagradu. Na svu sreću pogrešno prepíše najviše jednu *desnu* zagradu, a sve ostale prepíše točno. Tu nastaju problemi za Ivicu, jer ga potpuno zbuni ta zagrada koju je Marica krivo prepisala.

Tvoj zadatak je pomoći Ivici i napisati program koji će ispisati poziciju krivo postavljene *desne* zgrade u zadanom matematičkom izrazu.

Napomena: Nije potrebno voditi računa o prioritetu zagrada, tj. moguće je da npr. (dođe prije { i sl.

Ulazni podaci:

- string **st** koji predstavlja matematički izraz i neće imati više od 50 znakova.

String **st** može sadržavati:

- o znamenke dekadskog brojevnog sustava,
- o znakove osnovnih aritmetičkih operacija (+, -, *, /),
- o zagrade (*lijeve* i *desne*).

String **st** će uvijek imati jednak broj otvorenih i zatvorenih zagrada, ili uopće neće imati zagradu.

Za svaku poziciju u stringu **st** vrijedi da je lijevo od nje broj *lijevih* zagrada veći ili jednak broju *desnih* zagrada.

Izlazni podaci:

- ukupni broj zagrada u izrazu **st** (*lijevih* i *desnih*);
- pozicija pogrešno napisane *desne* zgrade u izrazu **st** (pozicija nekog znaka u izrazu je redni broj tog znaka u pripadnom stringu, pri čemu prvi znak u stringu ima redni broj 1, drugi 2, ...)

ILI

- 0 ako su sve zagrade ispravno zatvorene.

Primjeri:

RB	Ulaz	Izlaz
1.	$2+3*(2+5)$	2 0
2.	$-\{1+2*(3+1)-10\}+5$	4 15
3.	$1+(2+[3+\{4+5\}])$	6 13

• Napiši program u Pythonu:

Učiteljica se jako naljutila na Mariju i Ivanu kada ih je uhvatila da igraju križić-kružić za vrijeme sata. Kako bi ih naučila da se više ne igraju na satu, zadala im je jednu veliku riječ i rekla da:

- prebroje sve **podriječi** koje se nalaze u njoj,
- prebroje sve **jedinstvene podriječi**, tj. **podriječi** koje se pojavljuju samo jednom,
- nađu najdulji **palindrom** među **podriječima**, (onaj koji ima najviše slova). Ukoliko postoji dva ili više **palindroma** koji su "najdulji" traženi je onaj koji je **prvi** po abecedi.

Podriječ zadane riječi je niz od jednog ili više susjednih slova koja se nalaze u istom redosljedu kao u zadanoj riječi. **Podriječ** zadane riječi je i sama zadana riječ.

Palindrom je riječ koja se jednako čita s lijeva na desno i s desna na lijevo, npr. *KISIK*, *KRK*, *ANA*, itd. Pomozite učiteljici i napišite program koji će provjeriti jesu li Marija i Ivana dobro riješile zadatak.

Ulazni podaci:

- riječ koja se sastoji samo od velikih slova engleske abecede i koja će uvijek imati više od 0 i manje od 20 znakova.

Izlazni podaci:

- tekst "Ukupno podrijeci:" i iza njega prirodni broj **N**, ukupan broj **podriječi** u zadanoj riječi;
- tekst "Jedinstvenih podrijeci:" i iza njega prirodni broj **K**, broj **jedinstvenih podriječi** u zadanoj riječi,
- tekst "Najdulji palindrom:" i iza njega traženi **palindrom**.

Primjeri:

RB.	Ulaz	Izlaz	Komentar
1.	MAMA	Ukupno podrijeci: 10 Jedinstvenih podrijeci: 4 Najdulji palindrom: AMA	<i>Podriječi: M, A, M, A, MA, AM, MA, MAM, AMA, MAMA</i> <i>Jedinstvene: AM, MAM, AMA, MAMA</i>
2.	BANANA	Ukupno podrijeci: 21 Jedinstvenih podrijeci: 10 Najdulji palindrom: ANANA	

Zadatak 1

- Kojom naredbom ćemo dobiti binarnu vrijednost dekadskog broja?

1. `bin`

2. `oct`

3. `hex`

Zadatak 4

- Kojom naredbom ćemo pronaći ASCII kod znaka?

1. `chr`

2. `ord`

3. `int`

Zadatak 5

- Ivana ima X kuna, a cijena proizvoda kojeg želi kupiti je Y . Kako ćemo dobiti broj koliko proizvoda Ivana može kupiti?
 1. X / Y
 2. $X // Y$
 3. $X \% Y$

Zadatak 6

- Ivana ima X kuna, a cijena proizvoda kojeg želi kupiti je Y . Ako Ivana kupi maksimalno proizvoda koliko može s novcem kojeg ima, kako ćemo dobiti koliko joj je novca ostalo?
- $X // Y$
 - X / Y
 - $X \% Y$

Zadatak 7

- Koja vrijednost je rezultat sljedećeg aritmetičkog izraza: $4 / 2$
1. 2
 2. 2.0
 3. "2.0"

Zadatak 9

- Kako ćemo provjeriti je li vrijednost varijable **a** paran broj?

1. $a \% 2 == 0$

2. $a \% 2 != 0$

3. $a \% 2 = 0$

Zadatak 10

- Ocjena odličan daje se učeniku ako je broj bodova u intervalu od 90-100. Kojim uvjetom ćemo provjeriti je li broj bodova u tom intervalu?
1. `broj_bodova < 90 or broj_bodova > 100`
 2. `broj_bodova >= 90 and broj_bodova <= 100`
 3. `broj_bodova > 89`

Zadatak 12

- Kojim uvjetom ćemo provjeriti je li broj n djelitelj broja m ?
 1. $m \% n == 0$
 2. $n \% m == 0$
 3. $n == m$

Zadatak 13

- Potrebno je unositi s tipkovnice broj sve dok ne zadovoljava uvjet ($N < 100$). Koju petlju ćemo koristiti?

1. `while`
2. `for`

Zadatak 15

- Potrebno je ispisati sva slova od A do Z. Koju petlju ćemo koristiti?

1. `while`
2. `for`

Zadatak 16

- Potrebno je izračunati sumu prvih N prirodnih brojeva, N se unosi s tipkovnice. Na koju vrijednost ćemo inicijalizirati sumu?

1. 0

2. 1

3. N

Zadatak 17

- Svaka funkcija se sastoji od:

1. Tijela
2. Repa
3. Glave

Zadatak 18

- Mora li funkcija primiti parametre?

1. Da
2. Ne

Zadatak 19

- Mora li funkcija vraćati vrijednost?

1. Da
2. Ne

Zadatak 20

- Koju naredbu koristimo kako bismo iz funkcije vrijednost vratili u glavni program?

1. `return`
2. `print`
3. `break`

Zadatak 22

- Kojom gotovom funkcijom za rad sa stringovima ćemo sva slova “A” zamijeniti za slovo “E”?

1. find
2. replace
3. strip

Zadatak 23

- Kojom gotovom funkcijom za rad sa stringovima ćemo dobiti duljinu niza znakova?

1. `len`
2. `count`
3. `max`

Zadatak 24

- Kojom operacijom možemo iz niza znakova izvući potreban podniz?

1. Indeksiranje
2. Povezivanje
3. Uvišestručavanje

Zadatak 28

- $5 // 7$ je ...

Zadatak 29

- $5 \% 7$ je ...

Zadatak 30

- $4 // 2$ je ...