

Ime i prezime	Broj indeksa

**Obavezan deo A (15 poena – obavezno 10 poena):**

1. (2 poena) Za dati spisak odeljenja jedne škole koja su se plasirala za naredni krug školskog turnira u košarci, kao i spiskove odeljenja kojima fizičku kulturu predaju nastavnica Ana i odeljenja kojima predaje nastavnik Nikola, korišćenjem **operacija za rad sa skupovima** odredi koliko među plasiranim ima odeljenja kojima **NE** predaju ovo dvoje nastavnika?

```
ana = {'V-3', 'V-1', 'VI-1'}
```

```
nikola = {'VII-5', 'VII-2'}
```

```
plasirani = {'V-1', 'VII-5', 'VIII-5', 'V-3', 'VII-2', 'VI-3', 'VIII-4', 'VI-1'}
```

---

```
print(k)
```

- 2.
- a. (1 poena) Napisati funkciju **Cifre** u Python-u koja prihvata jedan ceo broj **x** i vraća true u slučaju da su sve cifre broja neparne. U suprotnom vraća false.
- b. (1 poena) Napisati program koji od korisnika traži da redom unosi cele brojeve **x** sve dok ne unese broj koji nema sve neparne cifre. Ispitivanje da li broj ima sve neparne cifre vršiti korišćenjem funkcije **Cifre**. Nakon što korisnik unese broj koji nema sve neparne cifre, program ispisuje koliko je uneto brojeva koji su imali sve neparne cifre.
3. (4 poena) Napisati rekurzivnu funkciju `novaL(L)` kojom se od date liste `L` pravi nova lista koja sadži elemente liste `L` bez ponavljanja (na pr. `novaL([1,2,3,4,5]) -> [1,2,3,4,5]`, `novaL([1,1,2,1,2,3]) -> [1,2,3]`)
4. (3 poena) Napisati program kojim se određuju dužine stranica trouglova  $a$ ,  $b$ ,  $c$ , takve da su  $a$ ,  $b$  i  $c$  prirodni brojevi za koje važi da je  $a \leq b \leq c$  (važi da je razlika dve stranice manja od treće, a zbir dve stranice veći od treće stranice) i da je obim trougla  $a + b + c$  jednak datom prirodnom broju  $O$ . (Rešiti na optimalan način)
5. (4 poena) Odrediti poslednji element  $n$ -tog reda trougaonog ispisa (na pr.  $n=3 \rightarrow 10236$ )? (Izvedena opšta formula tog elementa donosi 4 poena, u suprotnom zadatak nosi 2 poena)

```
1 6
16 36 76 156
316 636 1276 2556 5116 10236
.
```

## Izborni deo B (15 poena):

1. Organizuje se licitacija automobila zaplenjenih od strane carine. Postoje tačno tri automobila: Tiguan, Volvo i Reno. Učesnici licitacije svoje ponude šalju šifrovano u vidu binarnog zapisa. Ponuda jednog učesnika licitacije uvek ima tri dela (redom za date automobile). Ukoliko neki učesnik ne želi da da ponudu za neki od automobila, njegova ponuda za taj automobil je 0 (nula). Ponude su uvek celi brojevi.

U fajlu **Ponude.xml** dati su podaci o ponudama svakog učesnika licitacije (pogledati primer fajla).

Napisati program koji vrši organizaciju prispelih ponuda na sledeći način:

- Za svaku vrstu automobila se pamti iznos ponude i broj ponavljanja te iste ponude
- U fajl **SortiranePonude.json** ispisati prvo sve ponuda za Tiguana sortirane od najveće ka najmanjoj, zatim za Volvo i na kraju za Reno.
- Za svaki automobil na standardni izlaz ispisati najveću **jedinstvenu** ponudu.

Ponude.xml	SortiranePonude.json
<pre>&lt;xml&gt;   &lt;ponuda&gt;     &lt;ucesnik&gt;Marko&lt;/ucesnik&gt;     &lt;iznosi&gt;       &lt;iznos&gt;         100111000100       &lt;/iznos&gt;       &lt;iznos&gt;         101110111000       &lt;/iznos&gt;       &lt;iznos&gt;         0       &lt;/iznos&gt;     &lt;/iznosi &gt;   &lt;/ponuda&gt;   &lt;ponuda&gt;     &lt;ucesnik&gt;Ana&lt;/ucesnik&gt;     &lt;iznosi&gt;       &lt;iznos&gt;         101110111000       &lt;/iznos&gt;       &lt;iznos&gt;         100111000100       &lt;/iznos&gt;       &lt;iznos&gt;         1111101000       &lt;/iznos&gt;     &lt;/iznosi &gt;   &lt;/ponuda&gt;   &lt;ponuda&gt;     &lt;ucesnik&gt;Petar&lt;/ucesnik&gt;     &lt;iznosi&gt;       &lt;iznos&gt;         101110111000       &lt;/iznos&gt;       &lt;iznos&gt;         0       &lt;/iznos&gt;       &lt;iznos&gt;         1111101000       &lt;/iznos&gt;     &lt;/iznosi &gt;   &lt;/ponuda&gt;   &lt;ponuda&gt;     &lt;ucesnik&gt;Ivana&lt;/ucesnik&gt;</pre>	<pre>[   {     "auto": "Tiguan",     "iznos": 3000,     "broj": 3   },   {     "auto": "Tiguan",     "iznos": 2500,     "broj": 1   },   {     "auto": "Volvo",     "iznos": 4000,     "broj": 1   },   {     "auto": "Volvo",     "iznos": 3000,     "broj": 1   },   {     "auto": "Volvo",     "iznos": 2500,     "broj": 2   },   {     "auto": "Reno",     "iznos": 1000,     "broj": 2   } ]</pre>

```
<iznosi>
  <iznos>
    101110111000
  </iznos>
  <iznos>
    100111000100
  </iznos>
  <iznos>
    0
  </iznos>
</iznosi >
</ponuda>
<ponuda>
  <ucesnik>Una</ucesnik>
  <iznosi>
    <iznos>
      0
    </iznos>
    <iznos>
      111110100000
    </iznos>
    <iznos>
      0
    </iznos>
  </iznosi >
</ponuda>
</xml>
```