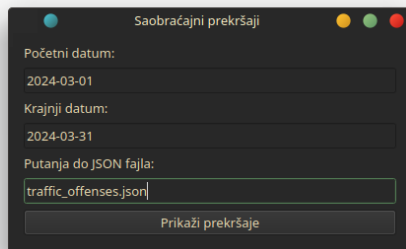
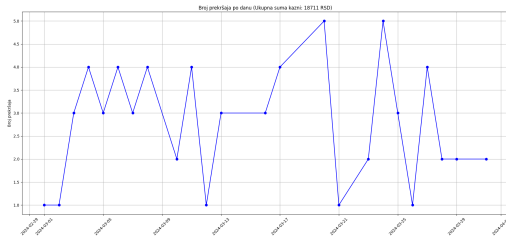


## R smer, Dizajn Programskih Jezika, Jun 2024

1. **[Python]** Napisati program koji omogućava korisniku da unese vremenski interval i putanju do JSON fajla koji sadrži podatke o saobraćajnim prekršajima. Svaki prekršaj je predstavljen objektom sa poljima: datum, prekršaj i novčana kazna. Program treba da nacrtati linijski dijagram koji prikazuje broj prekršaja za svaki dan u zadatom vremenskom intervalu, dok u naslovu grafika treba prikazati ukupnu sumu novčanih kazni za sve prekršaje u tom intervalu. Napraviti odgovarajući grafički interfejs sa dva polja za unos datuma (početak i kraj intervala) i jedno polje za unos putanje do JSON fajla, kao i dugme za potvrdu unosa. Klikom na dugme prikazuje se dijagram sa traženim podacima. Ukoliko je otvaranje JSON datoteke neuspešno, prikazati odgovarajuću poruku.



Slika 1: Interfejs sa poljima za unos



Slika 2: Prikaz informacija nakon klika na dugme "Izracunaj"

2. **[Go]** Implementirati konkurentan program koji obradom n-dimenzionalnog vektora transformiše svaki broj unet sa standardnog ulaza. Korisnik unosi broj brojeva, zatim te brojeve. Svaki uneti broj u vektoru se obradi na način da se svaka cifra u broju transformiše tako što se oduzme od najveće cifre u tom broju. Nakon toga, cifre transformisanog broja treba obrnuti kako bi se dobio konačan rezultat. Program treba da koristi zasebne gorutine za obradu svakog broja, pri čemu broj gorutina treba da odgovara broju dostupnih procesora. Nakon obrade svih brojeva, program treba da ispiše transformisane brojeve na standardni izlaz.

Napomena: *Zadatak se može rešiti sinhronizacijom niti preko katanca ili bez sinhronizacije, podelom poslova po nitima unapred. Pretpostaviti da su ulazni podaci ispravni.*

Primer 1

```
Ulaz:
3
321 5869 327
Izlaz:
210 314 54
```

Primer 2

```
Ulaz:
5
12345 67890 98765 13579 24680
Izlaz:
1234 90123 43210 2468 80246
```

Primer 3

```
Ulaz:
4
345 456 789 102
Izlaz:
12 12 12 21
```

Primer 4

```
Ulaz:
6
0 1 10 99 210 1000
Izlaz:
0 0 10 0 210 1110
```

3. **[Go]** Letnje vreme je idealno za organizaciju istraživačkih ekspedicija. Četiri istraživača su odlučila da krenu na istraživanje u neistraženi deo tropske džungle. Svako od njih treba da prikupi određene vrste uzoraka: biljne, insektne, mineralne ili životinjske. Na terenu se nalaze četiri istraživačka stajališta, od kojih svako nudi samo jedan tip uzorka. Svako stajalište ima kapacitet od 500 jedinica i periodično se dopunjuje novim resursima svakih 5 sekundi.

Istraživači kreću iz nasumično odabranih stajališta (dozvoljeno je da više njih krene iz istog stajališta) i redom odlaze na sledeće stajalište kako bi prikupili sve potrebne uzorke. Kada svi istraživači prikupe sve potrebne uzorke, program se zaustavlja.

**Stajališta:** Svako stajalište ima jedinstveni identifikacioni broj, trenutno stanje, maksimalni kapacitet i oznaku B, I, M ili Z u zavisnosti od vrste resursa koji se pruža.

**Istraživači:** U slučaju da stajalište trenutno nema dovoljno uzoraka, istraživač mora čekati dok se zalihe ne obnove. Kada završi sa prikupljanjem uzoraka u jednom stajalištu, istraživač prelazi u sledeće nasumično odabrano stajalište. Potrebno je kreirati četiri istraživača i inicijalno ih postaviti na nasumično odabrana stajališta. Takođe, inicijalno, njihovi rančevi su prazni. Svakog istraživača treba implementirati kao zasebnu gorutinu. Pri kreiranju istraživača proslediti potrebne količine biljaka, insekata, minerala ili životinja (nasumično izabrani brojevi od 0 do 500 koji se odnose na broj resursa odgovarajuće vrste).

Modelovati opisan sistem prikupljanja resursa i prikazivati informacije kada neki istraživač uđe u stajalište, kada završi prikupljanje uzoraka i, na kraju, kada neki istraživač završi sa prikupljanjem svih potrebnih uzoraka.

4. [C++] Napisati šablonsku klasu *Niz* koja predstavlja n-dimenzionalni niz. Svi podaci su tipa T koji je parametar šablona. U klasi obezbediti:

- (a) Konstruktor koji prima niz vrednosti.
- (b) Metod **normalizuj()** koji normalizuje niz ako norma niza nije 0.
- (c) Preopterećenje operatora << za ispis svih komponenti niza u formatu [x1, x2, x3, ..., xn] na proizvoljni izlazni tok.

Napisati program koji kreira niz brojeva u pokretnom zarezu tako što korisnik unese broj elemenata i njihove vrednosti, normalizuje taj niz i potom ga ispiše na standardni izlaz.

Primer 1	Primer 2	Primer 3	Primer 4
<pre> Ulaz: 3 3.0 4.5 7.3 Izlaz: [0.330209, 0.495314, 0.803509]</pre>	<pre> Ulaz: 3 3 4 5.0 Izlaz: [0.424264, 0.565685, 0.707107]</pre>	<pre> Ulaz: 2 5 7.1 Izlaz: [0.575779, 0.817606]</pre>	<pre> Ulaz: 3 0 0.0 0 Izlaz: [0, 0, 0]</pre>

5. [C++] Napisati program za praćenje sprovođenja projekata u kompaniji.

- (a) Napisati klasu *Projekt* koja sadrži podatke o projektu: naziv (string), budžet (neoznačen decimalni broj) i datum završetka (neoznažen ceo broj koji predstavlja broj dana od današnjeg).
- (b) Napisati klasu koja predstavlja funkcijski objekat *NajboljiProjekt* koja pri konstrukciji pamti datum završetka i budžet trenutnog najboljeg projekta inicijalno postavljen na maksimalne ili nevažeće vrednosti, kao i maksimalni dozvoljeni budžet. Pri funkcijskom pozivu ovog objekta se prosleđuje instanca klase *Projekt* i proverava se da li je taj projekat bolji u smislu najskorijeg datuma završetka ili istog datuma ali sa većim budžetom koji ne prelazi maksimalni budžet. Ako jeste, ažurira se vrednost funkcijskog objekta za trenutni najbolji projekat.
- (c) Napisati šablon funkciju koja za argumente dobija proizvoljnu kolekciju objekata klase *Projekt* i maksimalni dozvoljeni budžet i ispisuje naziv projekta sa najskorijim datumom završetka, ili ako više projekata imaju isti datum, projekat sa najvećim budžetom koji ne prelazi dozvoljeni.
- (d) Sa standardnog ulaza se učitava ukupan broj projekata, podaci o njima i maksimalni dozvoljeni budžet, i ispisuje naziv najboljeg projekta prema kriterijumima.

Primer 1	Primer 2	Primer 3	Primer 4
<pre> Ulaz: 3 Quantum 15000 200 Eclipse 20000 150 Nebula 18000 175 20001 Izlaz: Eclipse</pre>	<pre> Ulaz: 4 Aurora 22000 100 Zenith 21000 250 Vertex 25000 120 Phoenix 23000 150 21500 Izlaz: Zenith</pre>	<pre> Ulaz: 3 Fusion 18000 200 Nebula 20000 200 Eclipse 21000 180 20000 Izlaz: Nebula</pre>	<pre> Ulaz: 2 Vortex 14000 160 Pinnacle 14000 145 10000 Izlaz: Nema adekvatnog projekta!</pre>