

cctc 2019

coding contest Tuzla canton

17.maj 2019.

JU MS hemijska škola Tuzla

BILTEN

PRVO KANTONALNO TAKMIČENJE
U PROGRAMIRANJU ZA UČENIKE
OSNOVNIH ŠKOLA U
TUZLANSKOM KANTONU



AGENDA TAKMIČENJA

VRIJEME	AKTIVNOSTI
09:00 – 09:30	Registracija takmičara
09:30 – 10:00	Otvaranje takmičenja
10:00 – 13:00	Takmičenje
10:00 – 11:00	Okrugli sto – Iskustva sa takmičenja
13:00 – 13:30	Bodovanje i žalbe
13:30 – 14:00	Proglašenje pobjednika i dodjela nagrada

SPISAK POZVANIH TAKMIČARA

GRAD/OPĆINA	TAKMIČARI	ŠKOLA	MENTORI
TUZLA	Babić Borna	JU OŠ „Sjenjak“	Mehikić Zlatan
	Mahmutović Samir	JU OŠ „Pazar“	Spahić Damir
	Tukić Belmin	JU OŠ „Novi Grad“	Ljubović Sulejman
	Islamović Almir		
ŽIVINICE	Čačković Melika		
	Imamović Belmin		
	Jahić Haris		
	Husović Emira	JU OŠ Tušanj	Slomić Elvira
GRAČANICA	Imamović Ena		
	Džihanović Enes	JU OŠ Đurđevik	Butković Mirzeta
	Rahimić Elvis	JU OŠ Višća	Mešić Senada
GRADACAC	Osmanović Eldin	JU OŠ Šerići	Aličić Asmira
	Srabović Dina	Druga OŠ Gračanica	Djedović Maida
	Isić Dženan		
LUKAVAC	Zahirović Asmir	JU OŠ Džakule	Vikalo Mirnes
	Osmanović Bedrudin	OŠ „H. Kikić“ Vida	Isanović Dino
	Delić Harun		
SREBRENİK	Taletović Jasim	OŠ „M.Ć.Ćatić“ D.Zelinja	Halilović Nijaz
	Mekić Aldin	OŠ Dobošnica	Nuhanović Edin
	Ograšević Amir	OŠ „Lukavac Grad“	Bektić Edisa
KALESIJA	Salkić Nedim	OŠ Puračić	Mujkić Adeviya
	Zahirović Dženan	Prva OŠ Srebrenik	Mudrov Emina
	Jahić Rijad	Druga OŠ Srebrenik	Delić Nedim
	Softić Zuhdija	OŠ Duboki Potok	Sadić Alma
BANOVICI	Tursić Belmin		
	Škahić Enis	OŠ Rapatnica	Pjević Muhamed
	Alić Almedin	OŠ Rainci Gornji	Bajraktarević Adela
KLADANJ	Alić Senada	OŠ Vukovije	Jahić Asmir
	Majdančić Hasan		
DOBOJ ISTOK	Okanoović Emir	OŠ „Grivice“ Banovići	Saletović Zumreta
	Krekić Vedad		
	Gazdić Ado	OŠ Kladanj	Imamović Zijad
	Mujić Belmin	OŠ Klokočnica	Imširović Amela

ORGANIZACIJA TAKMIČENJA

Organizatori Prvog kantonalnog takmičenja iz programiranja za učenike osnovnih škola u Tuzlanskom kantonu su:

1. Pedagoški zavod Tuzla
2. JU MS hemijska škola Tuzla
3. Udruženje profesora i nastavnika informatike TK

Takmičenje se realizuje uz saglasnost Ministarstva obrazovanja i nauke TK.

Članovi Komisije za Prvo kantonalno takmičenje iz programiranja su:

1. Josip Vojnić, savjetnik iz Pedagoškog zavoda Tuzla
2. Enes Bećirović, ispred škole domaćina JU MS hemijska škola Tuzla
3. Nedžad Husić, ispred Udruženja profesora i nastavnika informatike u TK

Dežurni profesori na takmičenju su:

1. Aida Bajrić
2. Delila Mehmedinović
3. Ramiza Bećirović
4. Admira Hodžić

Tema okruglog stola „Iskustva sa takmičenja“

JU MS HEMIJSKA ŠKOLA TUZLA



Hemija škola u Tuzli je osnovana 1946. godine, a 31. mart uzet je kao Dan Hemijske škole Tuzla. Od 1965. godine škola se nalazi u sadašnjoj zgradi na adresi: ul Bosne Srebrene br. 8 u Tuzli. Zgrada i unutrašnjost škole je namjenski građena i opremljena prema standardima UNICEF-a za hemijsku struku. Sada nosi naziv Mješovita srednja hemijska škola u Tuzli, u kojoj se obrazuju učenici i polaznici za III, IV i V stepen složenosti.

Tokom godina razvoja škola je unapređivala i poboljšavala uslove za obrazovanje učenika kao i radne uslove zaposlenih. U Hemijskoj školi obrazuju se učenici za sljedeća stručna zvanja:

- hemijski tehničar - mikrobiolog
- hemijski tehničar nutricionist
- hemijski tehničar u farmaceutskoj proizvodnji
- grafički tehničar - web dizajner

Pored učenika u zanimanjima IV stepena planiran je i upis polaznika IV stepena kao i polaznika V stepena za sticanje zvanja majstora hemijsko tehnoloških procesa.

Planirano je obrazovanje 10 polaznika V stepena složenosti i 10 polaznika III i IV stepena složenosti.

JU MS HEMIJSKA ŠKOLA TUZLA

Nalazimo se u ulici **Bosne Srebrene broj 8**

telefon / fax: **035 / 281 - 092**

e-mail: **hemijska.skola@gmail.com**

web: **www.hemijaskkola.com.ba**



OKRUGLI STO „ISKUSTVA SA TAKMIČENJA“

Na današnjem takmičenju, paralelno sa izvođenjem takmičenja trajao je i okrugli sto koji su činili mentori učenika na temu „Iskustva sa takmičenja“ .

Prisutne je u temu uveo Nedžad Husičić navodeći opravdanost promovisanja programiranja u osnovnim školama zbog sve veće potrebe u današnjem dobu digitalne revolucije za afirmisanje programiranja u osnovnoj školi.

Ovo takmičenje je jedan oblik motivacije učenika i načina pronalaženja talenata u ovoj oblasti.

Sastanku je prisustvovao i savjetnik ispred Pedagoškog zavoda Tuzlanskog kantona Josip Vojnić, koji je prisutne upoznao o procesu stvaranja nastavnih planova i programa iz tehničkog odgoja i informatike kroz zajedničku jezgru za ove predmete a koje je objavila državna služba 2016. godine.

Na okruglom stolu je konstantovano da se na području Tuzlanskog kantona kreće sa pripremanjem za prelazak na jedan programski alat koji će se koristiti u svim ostalim školama a po mišljenju većine to bi trebao biti Python koji je po sintaksi najprilagodljiviji osnovnom školskom uzrastu.

Profesor Husičić je prisutne upoznao i sa problematikom sufinansiranja ovog događaja ali je iznio i optimizam da bi se ovo takmičenje moglo nastaviti i u narednim godinama u još većem obimu.

U narednom periodu Udruženje profesora i nastavnika informatike u Tuzlanskom kantonu planira organizovati seminare za edukaciju profesora i nastavnika u pogodnom programskom okruženju (Python), a o tome će profesori i nastavnici biti naknadno obaviješteni.

Opći utisak je da je takmičenje uspješno organizovano s obzirom da se kasno krenulo u organizaciju i da je naknadno uvršteno u kalendar takmičenja.

REZULTATI TAKMIČENJA

POZICIJA	TAKMIČARI	ŠKOLA	BODOVI
1	Gazdić Ado	OŠ "Kladanj"	25
2	Babić Borna	OŠ "Sjenjak"	20
3	Mahmutović Samir	OŠ "Sjenjak"	15
4	Imamović Belmin	OŠ "Novi Grad" Tuzla	13
5	Mujić Belmin	OŠ "Klokotnica"	11
6	Majdančić Hasan	OŠ Vukovije	10
7	Džihanović Enes	OŠ "Đurđevik"	10
8	Jahić Rijad	Druga OŠ Srebrenik	10
9	Zahirović Dženan	Prva OŠ Srebrenik	10
10	Rahimić Elvis	OŠ "Višća"	10
11	Husović Emira	OŠ "Novi Grad" Tuzla	10
12	Alić Almedin	OŠ "Rainci Gornji"	10
13	Imamović Ena	OŠ "Tušanj"	10
14	Srabović Dina	Druga OŠ Gračanica	10
15	Tursić Belmin	OŠ "Duboki Potok"	10
16	Taletović Jasim	OŠ "Musa Ćazim Ćatić" Zelinja Donja	10
17	Isić Dženan	Druga OŠ Gračanica	10
18	Osmanović Bedrudin	OŠ "Hasan Kikić" Gradačac	10
19	Alić Senada	OŠ "Vukovije"	10
20	Osmanović Eldin	OŠ "Šerići"	9
21	Softić Zuhdija	OŠ "Duboki Potok"	6
22	Ograšević Amir	OŠ "Lukavac Grad"	6
23	Čačković Melika	OŠ "Novi Grad" Tuzla	5
24	Jahić Haris	OŠ "Novi Grad" Tuzla	5
25	Delić Harun	OŠ "Hasan Kikić" Gradačac	5
26	Okanović Emir	OŠ "Grivice" Banovići	5
27	Salkić Nedim	OŠ "Puračić"	5

GALERIJA FOTOGRAFIJA









ZADACI SA TAKMIČENJA I RJEŠENJA

Zadatak: Plivanje

Na prvenstvu u plivanju nastupilo je N veslača. Zbog ograničenog broja staza za plivanje, a prema rezultatima iz kvalifikacija, plivači su nastavili takmičenje **podjeljeni u deset grupa**, tzv. finala. Finala su, prema važnosti, označena slovima A, B, C, D, E, F, G, H, I i J. U prvih devet finala (svi osim J) nastupa **jednak, najveći moguć** broj plivača, dok u J finalu nastupaju oni koji nisu raspoređeni u prvih devet finala. U nekim slučajevima J finale se i ne održi (kad u njemu nema takmičara).

Ukupna pozicija nekog takmičara na prvenstvu ovisi o **finalu** u kojem je nastupio te **poziciji** koju je zauzeo u njemu. Tako je npr. plivač koji je na prvenstvu od 60 plivača bio peti u C finalu, ukupno zauzeo 17. mjesto na prvenstvu (6 plivača iz A finala + 6 plivača iz B finala + 5 = 17).

Napiši program koji na temelju osvojene pozicije u finalu zadane oznake, određuje i ispisuje ukupnu **poziciju osvojenu na prvenstvu**. Napomena: ulazni podaci realno će opisivati moguće rezultate.

ULAZNI PODACI

U prvom redu nalazi se prirodan broj N ($10 \leq N \leq 100$), ukupan broj plivača na prvenstvu.

U drugom redu nalazi se jedan znak Z ('A' $\leq Z \leq$ 'J'), oznaka finala iz teksta zadatka.

U trećem redu nalazi se prirodan broj P ($1 \leq P \leq 10$), osvojena pozicija u finalu s oznakom Z .

IZLAZNI PODACI

U jednom redu treba ispisati ukupnu poziciju na prvenstvu iz teksta zadatka.

PRIMJERI TEST PODATAKA

ulaz	ulaz	ulaz
60	30	99
C	H	E
5	2	1
izlaz	izlaz	izlaz
17	23	45

Rješenje zadatka

Najvažniji dio zadatka je rečenica da u prvih devet finala nastupa **jednak, najveći moguć** broj plivača. To znači da se ukupan broj takmičara treba cjelobrojno podijeliti s devet. Na taj način ćemo saznati broj takmičara u svakom od prvih 9 finala. Za J finale nije potrebno odrediti broj takmičara u njemu jer zadatak garantira da će ulazni podaci realno opisivati moguće rezultate te će za to finale biti dovoljno gledati trenutnu poziciju takmičara u njemu.

Sada ostaje samo odrediti kako oznaku finala pretvoriti u brojčanu oznaku. To možemo uraditi naredbama odlučivanja ili korištenjem svojstava ASCII tablice.


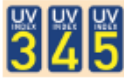



Programski kod (pisan u pseudojeziku)

```
ulaz (N);
ulaz (Z);
ulaz (P);
koliko_u_finalu := N div 9;
izlaz ((ord(Z)-65) * koliko_u_finalu + P)
// ord(Z) - redni broj znaka Z u ASCII tablici
```

Potrebno znanje: rad sa znakovima

Zadatak: UV

Tokom ljeta često se čuju upozorenja o štetnosti pretjeranog izlaganja Suncu i njegovom ultraljubičastom zračenju (kratica UV, *eng. ultraviolet*). **UV indeks** prirodan je broj koji pokazuje stepen opasnosti od UV zračenja. Što je on veći, veća je i opasnost od štetnog djelovanja. Za svaku vrijednost UV indeksa imamo **odgovarajuću poruku** koju ta vrijednost nosi. Znamo da vrijedi:

Oznaka					
UV indeks	1 i 2	3, 4 i 5	6 i 7	8, 9 i 10	11 i više
Poruka	Niska opasnost	Umjerena opasnost	Visoka opasnost	Vrlo visoka opasnost	Ekstremna opasnost

Napiši program koji za zadanu vrijednost UV indeksa ispisuje poruku koja se vezuje uz tu vrijednost.

ULAZNI PODACI

U prvom retku nalazi se prirodan broj **UV** ($1 \leq UV \leq 15$), vrijednost UV indeksa.

IZLAZNI PODACI

U jednom redu treba ispisati jednu od pet poruka navedenih u tekstu zadatka. To su „Niska opasnost“ ili „Umjerena opasnost“ ili „Visoka opasnost“ ili „Vrlo visoka opasnost“ ili „Ekstremna opasnost“. Poruke moraju biti ispisane u identičnom obliku kako su i definisane.

ulaz 2	ulaz 4	ulaz 9
izlaz Niska opasnost	izlaz Umjerena opasnost	izlaz Vrlo visoka opasnost

Rješenje zadatka

Za svaku od ponuđenih vrijednosti, naredbom odlučivanja treba definirati odgovarajući ispis. Samo je dodatno trebalo paziti i provjeriti što se događa kada UV indeks prijeđe preko vrijednosti 11.

Ovdje je dobar trenutak upozoriti na neophodnost ispisa ponuđenih poruka u identičnom obliku kako su zadane. Jer svaka automatska evaluacija putem Evaluatora za pogreške tog tipa neće imati razumijevanja.

Programski kod (pisan u psudojeziku)

```
ulaz (UV);  
ako je UV == 1 onda  
izlaz ('Niska opasnost');  
ako je UV == 2 onda  
izlaz ('Niska opasnost');  
ako je UV == 3 onda  
izlaz ('Umjerena opasnost');  
ako je UV == 4 onda  
izlaz ('Umjerena opasnost');  
ako je UV == 5 onda  
izlaz ('Umjerena opasnost');  
ako je UV == 6 onda  
izlaz ('Visoka opasnost');  
ako je UV == 7 onda  
izlaz ('Visoka opasnost');  
ako je UV == 8 onda  
izlaz ('Vrlo visoka opasnost');  
ako je UV == 9 onda  
izlaz ('Vrlo visoka opasnost');  
ako je UV == 10 onda  
izlaz ('Vrlo visoka opasnost');  
ako je UV >= 11 onda  
izlaz ('Ekstremna opasnost');
```

Potrebno znanje: naredba učitavanja i ispisivanja, naredba odlučivanja

Zadatak: Imenik

Jedan od načina na koji mobitelom možemo nazvati neki telefonski broj jest da taj broj utipkamo cifru po cifru te pritisnemo odgovarajuću tipku za poziv. Današnji pametni mobiteli u takvoj situaciji imaju jednu korisnu opciju. Nakon svake upisane cifre oni prikazuju sve brojeve zapisane u telefonskom imeniku koji počinju istim nizom cifara koje smo do tada utipkali. Npr:

Imenik		Telefonski broj koji želimo nazvati: 325643					
		3	2	5	6	4	3
234156	325643						
853256	421895						
325219	423900	325219	325219	325219	325643	325643	325643
754309	327896	325643	325643	325643			
218975	326100	327896	327896				
		326100	326100				

Ako su zadani svi telefonski brojevi zapisani u imeniku te broj koji želimo nazvati, napiši program koji će odrediti i ispisati koliko se brojeva iz imenika prikazalo nakon svake upisane cifre broja kojeg želimo nazvati.

Napomena: U imeniku mogu biti međusobno jednaki brojevi. U tom slučaju, svaki se gleda kao zaseban broj.

ULAZNI PODACI

U prvom retku nalazi se prirodan broj B ($100000 \leq B \leq 999999$), broj koji želimo nazvati.

U drugom retku nalazi se prirodan broj N ($1 \leq N \leq 10$), broj telefonskih brojeva u imeniku.

U sljedećih N redaka nalazi se po jedan prirodan broj TBi ($100000 \leq TBi \leq 999999$, $i=1..N$), i -ti telefonski broj u imeniku.

IZLAZNI PODACI

U prvi redat treba ispisati traženi broj brojeva iz imenika nakon što se upiše prva cifra broja B , u drugi red traženi broj brojeva nakon što se upiše druga cifra, u treći za treću cifru, u četvrti za četvrtu, peti za petu i u šesti redak broj prikazanih brojeva nakon upisane šeste cifre.

PRIMJERI TEST PODATAKA

ulaz	ulaz	ulaz
231456	985644	325643
3	5	10
237652	132457	234156
245567	985644	453256
100040	234321	325219
	789432	754309
	999999	218975
		325643
		421895
		423900
		327896
		326100
izlaz	izlaz	izlaz
2	2	4
1	1	4
0	1	2
0	1	1
0	1	1
0	1	1

Rješenje zadatka

Zadatak možemo riješiti posmatrajući zadane brojeve kao stringove, ali i kao prirodne brojeve. Ako se odlučimo za prirodne brojeve, trebamo osmisliti algoritam kojim možemo odrediti je li dva broja **počinju istim slijedom** znamenki. To je jednostavno odrediti ako se sjetimo da do vodećih znamenki u broju možemo doći gledajući količnik koji dobijemo kada cjelobrojno podijelimo taj broj s odgovarajućom potencijom broja 10.

Npr: za zadani broj 325643 vrijedi: $3 = 325643 \text{ div } 100000$; $32 = 325643 \text{ div } 10000$; $325 = 325643 \text{ div } 1000$; $3256 = 325643 \text{ div } 100$; $32564 = 325643 \text{ div } 10$; $325643 = 325643 \text{ div } 1$.

Kako je u zadatku definirano da će zadani brojevi biti šesteroznamenasti, tada je moguće i rješenje u kojem će se pojedinačno provjeriti svaka situacija. Ali, ovdje ćemo pokazati rješenje koje uključuje nizive i naredbu ponavljanja.

Programski kod (pisan u psudojeziku)

```
ulaz (B);
ulaz (N);
ulaz (imenik); // imenik je niz duljine N u kojem su zapisani telefonski brojevi
pot = 100000; // početna potencija od deset s kojom ćemo dijeliti brojeve
za i := 1 do 6 činiti // kako bi provjerili svaku kombinaciju početnog slijeda
{
    dio_od_B := B div pot; // početni slijed upisanih brojeva od broja B
    koliko := 0; // broj brojeva koji zadovoljavaju zadani uvjet
    za j := 1 do N činiti
    {
        dio_od_broja := imenik[j] div pot; //početni slijed od broja B
        ako je dio_od_B = dio_od_broja onda
            koliko := koliko + 1;
    }
    izlaz(koliko);
    pot := pot div 10;
}
```

Potrebno znanje: naredba odlučivanja, naredba ponavljanja, niz, algoritam određivanja početnog slijeda znamenki u broju

Zadatak: Isprika

Godinama su se putnici žalili da voz često kasni i ne dođe u krajnju stanicu u za to predviđeno vrijeme. Kako bi umirio putnike, Ministar saobraćaja se na **N dana** preselio na željeznički kolodvor i javno se izvinuo svaki put kada bi voz na toj relaciji **jako zakasnio**.

Ministar kaže da je voz iz polazne stanice jako zakasnio ako je **kasnio strogo više od prosjeka svih kašnjenja** vlakova iz polazne stanice prije njega.

Dok je živio na kolodvoru, Ministar je vodio evidenciju i sada želi na osnovi dobivenih podataka odgovoriti na sljedeća pitanja:

1. Koliko je bilo **ukupno kašnjenje** vozova u tih N dana izraženo u minutama;
2. Koji je dan, gledajući od dana Ministrova dolaska na kolodvor, voz **najviše kasnio**. Ako postoji više takvih dana, tada treba ispisati onaj koji je bio prije.
3. Koliko je **javnih isprika** zbog jakog kašnjenja voza Ministar uputio u tih N dana.

Napiši program koji će na osnovu zadanih ulaznih podataka ispisati odgovore na Ministrova pitanja. Napomena: Pretpostavljamo da je na prvi dan posmatranja dotadašnje prosječno kašnjenje vozova bilo nula minuta.

ULAZNI PODACI

U prvom redu nalazi se prirodan broj **N** ($1 \leq N \leq 20$), broj dana iz teksta zadatka.

U sljedećih **N** redaka nalaze se po četiri prirodna broja **S1, M1, S2 i M2** ($0 \leq S1, S2 \leq 23$, $0 \leq M1, M2 \leq 59$) odvojena razmakom, pri čemu su **S1 i M1** sat i minuta kada je vlak trebao doći u kolodvor, a **S2 i M2** su sat i minuta kada je vlak došao u kolodvor. Vrijeme predviđenog dolaska i stvarnog dolaska voza bit će zadani unutar istog dana. Voz nikad neće doći prije predviđenog vremena dolaska i barem jedan dan će zakasniti.

IZLAZNI PODACI

U prvi red treba ispisati prirodan broj, odgovor na prvo pitanje. U drugi red treba ispisati prirodan broj, redni broj dana koji predstavlja odgovor na drugo pitanje. U treći red treba ispisati prirodan broj, odgovor na treće pitanje.

Napomena: Tačan odgovor na neko od pitanja koji nije isписan u svom retku neće biti priznat.

PRIMJERI TEST PODATAKA		
ulaz	ulaz	ulaz
5	3	5
9 15 9 15	12 30 12 45	6 30 6 45
10 20 10 25	13 0 13 30	6 45 7 15
11 30 11 32	9 15 9 30	12 0 12 45
13 10 13 40		13 15 14 0
20 0 20 9		9 15 9 30
izlaz	izlaz	izlaz
46	60	150
4	2	3
2	2	4

Opis prvog test podatka:

Prvi dan: kašnjenje: 0 minuta; nema isprike ($0 \leq 0$), novo prosječno kašnjenje: 0 minuta;

Drugi dan: kašnjenje: 5 minuta; isprika ($5 > 0$); novo prosječno kašnjenje: 2.5 minuta;

Treći dan: kašnjenje: 2 minute; nema isprike ($2 < 2.5$); novo prosječno kašnjenje: 2.33 minute;

Četvrti dan: kašnjenje: 30 minuta; isprika ($30 > 2.33$); novo prosječno kašnjenje: 9.25 minuta

Peti dan: kašnjenje: 9 minuta; nema isprike ($9 < 9.25$); novo prosječno kašnjenje: 9.2 minuta

Rješenje zadatka

U ovom zadatku imali smo tri dijela. U takvim situacijama, ako nije moguće riješiti cijeli zadatak, treba se koncentrirati na to da se tačno riješe neki od dijelova cijelog zadatka. Npr. ukupno kašnjenje moglo se jednostavno riješiti na sljedeći način:

```
N = int(input())
kasnjenje_ukupno = 0
for i in range(N):
    S1, M1, S2, M2 = map(int, input().split())
    kasnjenje_ukupno += (S2 * 60 + M2) - (S1 * 60 + M1)
print(kasnjenje_ukupno)
```

Općenito, cjelokupno rješenje zadatka dobivamo pažljivim implementiranjem uslova koji se nalaze u tekstu zadatka.

Programski kod (pisan u Pythonu 3.4)

```
N = int(input()) # učitamo broj dana
# inicijaliziramo varijable
kasnjenje_ukupno = najvece = kada = prosjek = isprika = 0
for i in range(N): # ponovimo N puta
    S1, M1, S2, M2 = map(int, input().split()) # učitamo ulazne podatke
    # odredimo kašnjenje vlaka za i-ti dan
    kasnjenje_dan = (S2 * 60 + M2) - (S1 * 60 + M1)
    # ako je kašnjenje vlaka i-tog dana veće od prosjeka povećaj broj isprika
    if kasnjenje_dan >= prosjek:
        isprika += 1
    if kasnjenje_dan > najvece:
        najvece = kasnjenje_dan # zapamtimo to kašnjenje
        kada = i + 1 # zapamtimo koji se dan to dogodilo
    # povećamo ukupno kašnjenje za kašnjenje i-tog dana
    kasnjenje_ukupno += kasnjenje_dan
    # odredimo prosjek dosadašnjih kašnjenja
    prosjek = kasnjenje_ukupno / (i + 1)
print(kasnjenje_ukupno)
print(kada)
print(isprika)
```

Potrebno znanje: naredba ponavljanja, određivanje maksimalne vrijednosti