nedeljainformatike v5.0

OPŠTI TEST IZ RAČUNARSKIH NAUKA

18. septembar 2018. — 7. oktobar 2018.

Pre nego što počnete, pažljivo pročitajte ove instrukcije:

*Pred vama su četiri zadatka iz raznih oblasti računarskih nauka, za čiju izradu imate tačno **3 nedelje**. Kao deo prijave za Nedelju informatike, potrebno je da, pored popunjenih osnovnih podataka, pošaljete vaša rešenja za **bar jedan** zadatak. Ukoliko pošaljete rešenje za više od jednog problema, **najviše će se vrednovati najbolje urađen**. Ipak, savetujemo vas da probate da rešite što je moguće više zadataka.*

*Neki problemi su namerno **nedovoljno definisani** ili nemaju jedinstveno "tačno" rešenje. Potrebno je da, u okvirima koje postavlja zadatak, razvijete što bolje rešenje za takve probleme, sa obrazloženjem za odluke ili pretpostavke koje ste napravili.*

*Preporučujemo da odabrane probleme rešavate **što je kompletnije moguće**. U okviru jednog zadatka podzadaci nisu nužno sortirani po težini.*

*Uzimajući u obzir datu količinu vremena, dati problemi nisu jednostavni, i očekivanje je da će solidan kandidat uspešno rešiti **neznatno više od polovine problema koje je odabrao**.*

*Molimo vas da probleme rešavate **samostalno**! Međutim, dozvoljeno je korišćenje bilo kakve literature (što uključuje i Internet pretragu).*

Zadatke pripremili:

- **Andrej Ivašković**
- **Luka Jovičić**
- **Momčilo Topalović**
- **Vladimir Milenković**

1 Jurcanje

Posmatrajmo naredni problem *begunca i policajaca*. Jedan begunac se kreće po gradu stalnom brzinom 1, dok policajci imaju cilj da ga uhvate i svih n njih ($n \geq 1$) se kreću stalnom brzinom a (gde je a pozitivan realan broj). Ulice grada nemaju nadvožnjake i podzemne prolaze – drugim rečima, grad može da se predstavi kao *konačan euklidski planaran povezan graf* (ako je ijedna od reči ovde nepoznata, brza Internet pretraga pojma *Euclidean graph* će razjasniti). I policajci i begunac mogu isključivo u čvorovima da izvrše jednu od dve operacije:

- čekaj vreme T (gde je T pozitivan realan broj);
- kreni da se krećeš ga čvoru v (gde je v čvor sused trenutnom čvoru).

Voditi računa da je T broj o kom se odlučuje onda kada počinje izvršavanje operacije, tako da nije moguće "reagovati" na događaje.

I policajci i begunac u svakom trenutku poseduju aktuelne informacije o položajima i aktuelnom pravcu i smeru kretanja svih ostalih. Međutim, ukoliko neki policajac trenutno čeka, begunac nema informaciju koliko će još dugo čekati; takođe, ako begunac čeka, nijedan policajac ne zna koliko će još dugo čekati. Begunac ne počinje ni u jednom čvoru u kom se prvo nalazi neki policajac.

(*a*) Odrediti graf u kom za $a = 1$ i $n = 2$ policajci nemaju strategiju da uhvate begunca kakva god bila startna pozicija, ali zato imaju ako $a = 1$ i $n = 3$ za svaku startnu poziciju.

(*b*) Dokazati ili opovrgnuti sledeća tvrđenja:

(*i*) ako je $a > 1$, tada je dovoljan jedan policajac da bi se uvek uhvatio begunac;

(*ii*) za sve $a > 0$ i za svaku mapu grada, postoji n tako da za sve moguće startne pozicije policajaca i begunca, policajci imaju strategiju da uhvate begunca (ako je odgovor potvrdan, dodatno će se vrednovati rešenja sa manjim konkretnim n);

(*iii*) postoji neko $n \geq 1$ takvo da će policajci uvek imati strategiju kojom mogu da uhvate begunca, bez obzira na sve druge parametre (ako je odgovor potvrdan, dodatno će se vrednovati rešenja koja konstruišu najmanju vrednost n).

Rešenje ovog zadatka treba dostaviti u vidu `pdf` fajla sa objašnjenjima. Dozvoljeno je predati i skenirano (ili fotografisano) rukom ispisano rešenje **dokle god je ono čitko**. Skice su poželjne, ali ne i obavezne.

2 Protokoli

Jedan od osnovnih problema u kriptografiji se odnosi na to kako dve osobe, Alisa (A) i Bob (B) mogu bezbedno razmenjivati poruke. Ovaj zadatak daje nekoliko primera takvih protokola. Vaš zadatak će biti da ih 'razbijete'.

Na adresi `https://csweek.mg.edu.rs/static/resources/protocols.zip` se nalazi arhiva koja će vam biti potrebna za rešavanje nekih potproblema. Radi jednostavnosti, uvodimo sledeće tri oznake:

- Bitovno ekskluzivno ili dva broja x i y, u oznaci $x \oplus y$ – u mnogim programskim jezicima pisano kao $\hat{}$ ili **xor**.

- *Rolling hash* nekog niza ASCII karaktera (stringa) $(s_i)_{i=1}^n$, u oznaci $H_{x,p}$ predstavlja broj

$$H_{x,p}\left((s_i)_{i=1}^n\right) = \left(\sum_{i=1}^{n} s_i x^i\right) \bmod p.$$

  Primetimo da je numerička vrednost i-tog karaktera jednaka njegovoj vrednosti u ASCII tabeli – `asciitable.com`.

- Slučajni generator koji koristi osoba X, u oznaci $\mathcal{R}_X$, predstavlja pseudo-funkciju koja ne prima nikakav parametar, ali vraća vrednost $h_{X,i}$ kada je pozvana i-ti put, gde važi sledeća rekurentna veza:

$$h_{X,i} = (C_X + h_{X,i-1}) \bmod p,$$

  gde je p neki prost za koga važi $2^{4097} \leq p < 2^{4098}$. Napomenimo da su vrednosti $h_{X,0}$, kao i C_X jedino poznate X-u.

U ovom zadatku pretpostavljamo da svaki potencijalni napadač može da pročita svaku poruku. Za više informacija u vezi sa notacijom korišćenom u primerima (a) i (c) pogledajte prezentaciju o računarskoj bezbednosti sa prve nedelje informatike (`csnedelja.mg.edu.rs/static/resources/v1.0/security.pdf`).

U svakom od podzadataka predložite neku izmenu protokola koja uklanja ranjivost (ako takva izmena postoji).

(*a*) U ovom potproblemu, A i B razmenjuju N brojeva $m_i < n \approx 2^{4096}$. Oni se prvo dogovore oko jednog broja x koristeći server S, nakon čega salju poruku $m_i \oplus x$. Detaljniji opis protokola:

$$\begin{aligned} A: &\quad a_i \leftarrow \mathcal{R}_A \\ A \to S: &\quad B, a_i^3 \bmod n \\ S \to B: &\quad A \\ B: &\quad b_i \leftarrow \mathcal{R}_B \\ B \to S: &\quad A, b_i^3 \bmod n \\ S \to A: &\quad B, a_i \oplus b_i \\ A \to B: &\quad m_i \oplus b_i. \end{aligned}$$

Vaš zadatak je da opišete potencijalni napad koji bi dva napadača C i D mogla da izvedu kako bi otkrili **što više poruka** m_i. Pretpostavka je da C i D mogu da pročitaju, ali ne i da izmene, poruke između A, B i S.

(b) U ovom potproblemu, A i B razmenjuju stringove koristeći unapred dogovoreni ključ $0 \leq x < 128$. Svaka poruka $(s_i)_{i=1}^n$ se enkriptuje koristeći x na sledeći način:

$$A \to B: \quad (s_i \oplus x)_{i=1}^n.$$

U fajlu `subtask_a.in` se nalazi niz komunikacija (enkriptovanih poruka) između A i B. Poznato je da je svaka poruka enkriptovana koristeći isto x, kao i da su originalne poruke smislene rečenice na engleskom jeziku. Vaš zadatak je da napište program koji nalazi to x.

(c) U ovom potproblemu A šalje B-u poruku m tako što A prvo pošalje B-u zahtev za komunikaciju, na šta B odgovara javnim ključem koji A-u služi za enkripciju. Detaljniji opis protokola:

$$
\begin{aligned}
A \to B: \quad & A \\
B: \quad & \textbf{do}\ \{ \\
& \qquad p \leftarrow \mathcal{R}_B \\
& \}\ \textbf{while}\ (\lfloor \log_2 p \rfloor \neq 4095\ \textbf{or not}\ \text{prime}(p)); \\
& \textbf{do}\ \{ \\
& \qquad g \leftarrow \mathcal{R}_B \\
& \}\ \textbf{while}\ (g \geq p); \\
& x \leftarrow \mathcal{R}_B \\
& y \leftarrow g^x \bmod p \\
B \to A: \quad & g, p, y \\
A: \quad & k \leftarrow \mathcal{R}_A \\
& c_1 \leftarrow g^k \bmod p \\
& c_2 \leftarrow m y^k \bmod p \\
A \to B: \quad & c_1, c_2
\end{aligned}
$$

Vaš zadatak je da odredite kako B može da otkrije m, kao i da opišete potencijalni napad kojim bi napadač E mogao da otkrije m. Pretpostavka je da E može da pročita, ali ne i da izmeni, poruke između A i B.

(*d*) U ovom potproblemu A i B ne enkriptuju svoje poruke, ali vrše autentikaciju. To rade na sledeći način. Pretpostavimo da A želi da pošalje poruku $S = (s_i)_{i=1}^n$. Tada on, zajdeno sa S, šalje i $H_{x,p}$, gde je x unapred dogovoreni broj koji je vama **nepoznat**, a p unapred dogovoreni prost broj koji je vama **poznat**. B proverava da je poruka stvarno potekla od A tako što on sam računa $h = H_{x,p}(S)$, koju poredi sa vrednošću koju je primio. U fajlu `subtask_d.in` se nalazi niz od $n = 84$ komunikacija dužine do 25 između njih. Vaš zadatak je da pošaljete poruku:

```
cmd=init_self_distruct$shell=freejensh$auth=chifu
```

Drugim rečima, vaš zadatak je da otkrijete heš vrednost prethodne poruke.

Vaš odgovor treba da bude predat u vidu arhive koja sadrži:

- Jedan `.txt` ili `.pdf` fajl koji sadrži rešenja primera (a) i (c), kao i kratak opis algoritama korišćenih u primerima (b) i (d).
- Izvorne kodove korišćene u primere (b) i (d).

3 Upis

U ovom zadatku, od vas se očekuje da napišete program koji simulira proces koji dobro poznajete i kroz koji ste prošli (ili ćete uskoro) – upis u srednje škole. Svi fajlovi koji su vam potrebni za ovaj poduhvat se nalaze u arhivi koju možete naći na `https://csweek.mg.edu.rs/static/resources/upis.zip`.

U fajlu `ucenici.txt` se nalaze podaci o svim učenicima koji su ove godine upisali državnu srednju školu u prvom krugu. U svakom redu se nalaze informacije o jednom učeniku. One su izdeljene u više celina kosom crtom (`/`). Prva celina se sastoji od šestocifrene šifre učenika i šifre blizanca ako ga ima (0 u suprotnom), razdvojene zapetama (`,`). Druga celina je tekstualna šifra upisanog smera. Naredne tri celine su (celobrojne) zaključne ocene za svaki predmet u poslednja tri razreda, razdvojene zapetama. Šesta celina sadrži maksimalno jedan karakter, i to `v` ako je učenik nosilac Vukove diplome, a ništa u suprotnom. U sedmoj celini se nalaze brojevi bodova na tri završna ispita, broj bodova dobijenih zbog uspeha na takmičenjima i broj bodova dobijen na račun afirmativnih mera (tim redom). Naredne celine predstavljaju listu želja i svaka sadrži šifru jednog smera i dodatni broj bodova za taj smer koji potiče od polaganog prijemnog ispita ili takmičenja koje se dodatno boduje za dati smer. Na primer, red:

```
917806,0/JBSK GA 4R10S/5,5,5/5,5,5,5/5,5,5,5/v/11.05,9.1,10.5,0.0,
0.0/JBSK GA 4R15S,164.5/JBSK GA 4R10S,164.5/JBSK GA 4R17S,164.5
/JBSK GA 4R99S,164.5/
```

označava učenika sa šifrom 917806, bez blizanca, koji je upisao smer JBSK GA 4R10S, ima sve petice, vukovac je, ima ukupno 30.65 bodova na završnom ispitu, nema bodova za afirmativne mere ni za takmičenja i ima četiri želje. Za svaku od želja, ima 164.5 dodatnih bodova za taj smer.

U fajlu `kvote.txt` se nalazi lista šifri smerova sa maksimalnim brojem učenika koji taj smer može da upiše. Svaki smer se nalazi u zasebnom redu i odvojen je od broja koji predstavlja kvotu zapetom. Oba fajla možete otvoriti bilo kojim tekstualnim editorom.

Da podsetimo, ocene nose ukupno 60 bodova (opšti uspeh sva tri razreda se vrednuje podjednako), a završni ispit u zbiru nosi 40 bodova, na koje se dodaju bodovi za takmičenja i afirmativne mere. Na tako izračunat broj bodova dodaje se broj bodova za svaki specijalan smer, koji ne moraju (i nisu) uvek isti, pa je moguće da učenik ulazi u rangiranje sa različitim brojevima bodova za svaku želju. Blizanci imaju identične liste želja, ali čak i ako nemaju iste bodove, oba blizanca ulaze u rangiranje sa bodovima onog boljeg. (Detaljniji propisi za upis blizanaca se mogu naći u stručnom uputsvu za upis: `tinyurl.com/strucno-uputstvo`.) Naravno, čak i ako neki učenik ima najviše bodova za, recimo, drugu želju, prioritet je upisati ga na prvu ako ima mesta, pa tek onda pokušati sa onima niže na listi.

U slučaju da više učenika ima isto bodova, prioritet imaju oni sa Vukovom

diplomom, zatim oni sa više dodatnih bodova za taj smer, onda oni sa više bodova na takmičenju i na kraju oni sa više bodova na završnom ispitu. Ako su učenici po svim parametrima identični, oboje se upisuju preko kvote. Za sve nedoumice koje imate, možete se konsultovati sa zvaničnim pravilnikom: `upis.mpn.gov.rs/Lat/Pravilnici` ili ličnim iskustvom. Podaci su, uz minimalne izmene koje ne menjaju rezultat simulacije, preuzeti sa `upis.mpn.gov.rs`. Napravljene izmene obuhvataju stvari poput sabiranja dodatnih bodova za takmičenja sa bodovima na prijemnom ispitu za specijalizovane smerove, izbacivanje želja iz liste želja za koje učenik ne ispunjava uslov (između ostalog, nepoložen prijemni ispit ili nedovoljno bodova za četvorogodišnje smerove) i korigovanje dodatnih bodova za filološka odeljenja (član 21 pravilnika). U većini slučajeva, ako unesete šifru učenika na veb portal, podaci koji su vam prikazani će biti isti kao i oni u fajlu.

Vaš zadatak je da napišete kod koji kreira novi fajl, `bodovi.txt`, koji ispisuje šifre smerova i broj bodova potreban za upis, ako je kvota popunjena, ili 0 u suprotnom. Svaki smer treba da stoji u zasebnom redu, a šifra smera da bude razdvojena zapetom od broja bodova (slično formatu fajla `kvote.txt`).

(*a*) Napišite simulaciju koja ignoriše bodove za svaki smer. Za ovaj podzadatak, smatrajte da učenik ima isti broj bodova za svaku želju.

(*b*) Uzmite u obzir i dodatne brojeve bodova za svaki smer pri simulaciji. Međutim, ako ste očekivali da kad ovu simulaciju otkucate tačno kao rezultat dobijete da je svaki učenik upisao onaj smer koji vam je dat u fajlu sa podacima, bićete razočarani kad shvatite da to nije slučaj. Koliko god se trudili, rezultati dobijeni simulacijom će se razlikovati od tačnih. Što nas dovodi do...

(*c*) Pronađite šta vam nedostaje. Podaci za par učenika su izostavljeni i od vas očekujemo da nam kažete njihove šifre. Simulacija će vam omogućiti da vidite za koje smerove vam "nešto ne štima", nakon čega treba da na `http://upis.mpn.gov.rs` nađete šifre učenika koji vam nedostaju u fajlu sa podacima. Garantujemo da je izostavljen samo mali jednocifren broj učenika. Obavezno u objašnjenju rešenja navedite šifre učenika kojih nema u fajlu.

Možete koristiti bilo koji programski jezik i biblioteku. Nije potrebno rešavati svaki podzadatak posebno, već je dovoljno da predate najtačniju verziju simulacije koju imate i, ako ste našli, što više šifara učenika koji nedostaju. U jednoj arhivi, potrebno je da predate kod koji izvršava simulaciju, uključujući sve pomoćne funkcije koje ste koristili za pronalaženje učenika koji fale i detaljno objašnjenje postupka simulacije i načina na koji ste pronašli tražene šifre.

4 ĆTVI+

Nakon ulaza Ćrbije u Ćevropsku Uniju i nuklearne katastrofe, godine 3018, u gradu Ćeogradu je na vlast došla pacovska organizacija ĆTVI+, koja se zalaže za jednakost među svim štakorima u ovom gradu. Na početku svoje vladavine, dotična organizacija je naišla na puno izazova koje mora da reši, a vaš zadatak je da im u tome pomognete što je moguće bolje.

(*a*) Na predstojećim izborima, postoje samo dva kandidata za pobedu – stranka ĆNS na čelu sa Ratatuljem, kao i stranka ĆPP na čelu sa Splinterom. U gradu, takođe, postoji tačno N glasača, i zna se da svaki glasač mora izaći na izbore i glasati za jednu od dve gore navedene stranke. U grad je došao štakorski mason Ći Fu, i on želi da otkrije identitete lidera dve stranke, ali, pošto su svi srećni pacovi izomorfni, Ći Fu ne može da razazna pacove između sebe. Ali, Ći Fu može da vrši istraživanja, i to sledećeg oblika: on može izabrati neki podskup pacova, i pitati ih za koga bi glasali. Pacovi u tom podskupu glasaju po sledećem principu:

- Ukoliko se u izabranom podskupu nalazi Ratatulj, a ne nalazi se Splinter, svi pacovi iz tog podskupa će glasati za Ratatulja.
- Ukoliko se u izabranom podskupu nalazi Splinter, a ne nalazi se Ratatulj, svi pacovi iz tog podskupa će glasati za Splintera.
- Ukoliko se u izabranom podskupu nalaze i Splinter i Ratatulj, njih dvojica će glasati uvek za sebe, a ostali pacovi mogu birati za koga će glasati.
- Ukoliko se u izabranom podskupu ne nalaze ni Splinter ni Ratatulj, svaki pacov u podskupu može birati za koga će glasati.

Posle glasanja svakog podskupa, Ći Fu dobija broj glasača koji su glasali za Splintera, kao i broj glasača koji su glasali za Ratatulja. Kako istraživanja koštaju po čak jedan sir, pomozite Ći Fu-u da otkrije identitete Splintera i Ratatulja u što je manje moguće istraživanja.

Imajte u vidu da su pacovi jako zla mala stvorenja, tako da će se ponašati tako da, ako postoji način glasanja u kome oni uspeju da sakriju lidere stranaka od Ći Fu-a, oni će glasati po tom principu.

(*b*) Kako bi namestio izbore, lider ĆNS-a je otišao kod drevnog pacovskog vrača Ćostradamusa kako bi saznao ukupan broj pacova u gradu. Ćostradamus, kao i svi ostali veliki vračevi, više voli da odgovara u zagonetkama. Konkretno, on odgovara na sledeći način.

Neka je posle deratizacije pacova 3012. godine u Ćeogradu ostalo samo $N \leq 10^{18}$ pacova. Ćostradamus odgovara samo na jednu vrstu pitanja, i

to: Ratatulj mu saopšti broj x, na šta mu mudri pacov odgovara sa $\frac{\lfloor \frac{N}{x} \rfloor}{N}$ u vidu neskrativog razlomka (vrati mu uređeni par brojioca i imenioca). Na primer, ako je $N = 10$ i $x = 4$, Ćostradamus će odgovoriti sa $\frac{1}{5}$.

Pošto Ćostradamus za svaki svoj odgovor traži po dva koluta kačkavalja, a Ratatulj je poznat po svojim merama štednje, on želi da otkrije broj N uz pomoć što manje upita. Pomozite mu u njegovoj nameri.

(c) Jedan od prvih zadataka našeg vođe je da sortira sve stanovnike ovog grada po njihovoj dužini, u milimetrima. Po evoluciji (koja, uzgred budi rečeno nije tačna), zna se da ne postoje dva pacova na svetu koji imaju iste dužine. Svi stanovnici Ćeograda su se poređali na novootvorenom mostu na Ćlaviji, i naš Vođa može koristiti samo jednu tehniku sortiranja: ciklično rotiranje.

Malo formalnije: Dat nam je niz $(a_i)_{i=1}^{N}$, i znamo da su svi elementi različiti. Operacija koja nam je dozvoljena je:

Izabrati broj K ($K \leq N$), i k brojeva $i_1, i_2, \ldots, i_K$ ($1 \leq i_j \leq N$) tako da među njima ne postoje dva ista. Nakon toga, element na poziciji i_1 ide na poziciju i_2, element na poziciji i_2 ide na poziciju i_3, ... Element sa pozicije i_K ide na poziciju i_1.

Potrebno je sortirati niz primenom što je moguće manje ovakvih operacija.

(d) Izbori su napokon završeni. Jedna od stvari koju je nova (ili stara) vladajuća koalicija obećala da će uraditi odmah po završetku izbora je to da naprave, naizgled nasumičan, poredak svih pacova u državi. Znamo da u državi ima N pacova, i obe koalicije imaju istu tehniku pravljenja poretka, samo ne nužno sa istim parametrima. Tehnika je:

Algorithm 1 Tehnika zamene

```
1:  procedure SHUFFLE(K)
2:      while K > 0 do
3:          a ← rand(1, n)
4:          b ← a
5:          while b = a do
6:              b ← rand(1, n)
7:          end while
8:          swap(pacov[a], pacov[b])
9:          K ← K − 1
10:     end while
11: end procedure
```

Napomena: rand(a, b) vraća prirodan broj iz intervala $[a, b]$, i svaki broj iz tog intervala ima jednaku verovatnoću da bude vraćen. Na početku su svi pacovi

bili sortirani po dužini i zna se da je njihov broj deljiv sa 6. *Pojašnjenje: Na početku imamo sortiranu permutaciju brojeva od 1 do N, dakle* $pacov[i] = i$ *za svako* i *u intervalu* $[1, N]$. Ono oko čega lideri stranaka nisu mogli da se dogovore je misteriozan parametar K. Naime, Splinter je želeo da izabere $k = \lfloor \frac{N}{2} \rfloor$, dok je Ratatulj želeo $k = \lfloor \frac{N}{2} \rfloor + 1$. Na kraju je objavljena samo konačna permutacija, a našeg Ći Fu-a zanima čija strategija je bila korišćena. Kako Ći Fu nije baš najbistriji, on od vas traži pomoć.

Rešenje treba da bude predato u vidu jednog `.pdf` fajla koji sadrži rešenja svih rešenih podzadataka. Zanimljive ideje i ideje koje vode ka rešenju će se takođe vrednovati.

KRAJ TESTA