

Programiranje 2 – popravni kolokvij, 3. 9. 2021.

Rezultati i uvidi: Rezultati u srijedu, 8.9.2021. navečer na webu. Informacije o uvidima bit će objavljene naknadno.

Upute: Na kolokviju je dozvoljeno koristiti samo pribor za pisanje i brisanje, te službeni podsjetnik. Kalkulatori, razne neslužbene tablice, papiri i sl., nisu dozvoljeni! **Mobitele isključite i spremite!** Sva rješenja napišite isključivo na papire sa zadacima, jer jedino njih predajete. Obavezno predajte sve papire sa zadacima, čak i ako neke zadatke niste rješavali. Ne zaboravite se **potpisati** na svim papirima! Skice smijete raditi i na drugim papirima koje će vam dati dežurni asistent/ica.

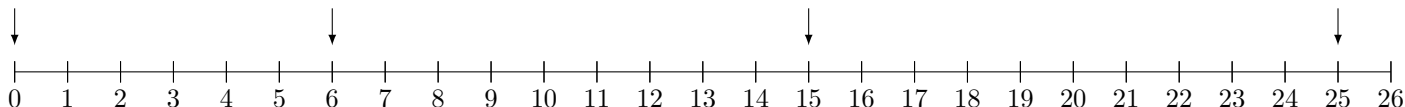
U svim zadacima **zabranjeno je korištenje dodatnih nizova** i standardne matematičke biblioteke (zaglavlje `math.h`), osim ako je u zadatku drugačije navedeno. Dozvoljeno je pisanje pomoćnih funkcija.

Uvjete navedene u zadacima (pozitivnost, ograde na n i sl.) ne treba provjeravati.

Zadatak 1 (20+5 bodova) Pat i Mat postavljaju $k \geq 1$ reklamnih panoa duž državne ceste D2021, čija duljina je ukupno n kilometara (pri čemu je n prirodan broj). Pritom moraju poštivati sljedeća pravila tvrtke za koju rade:

- Svaki pano mora biti postavljen na nekom cjelobrojnom kilometru ceste D2021. Pritom je prvi pano postavljen na 0-tom kilometru, a posljednji unutar posljednjih 4.5 kilometara ceste D2021.
- Svaki pano, osim prvog i zadnjeg, treba imati udaljenost od prethodnog panoa manju ili jednaku udaljenosti do sljedećeg panoa. Pritom udaljenost susjednih panoa mora biti minimalno 4 km.
- Zbog praznovjerja, ne postavljaju se panoi na p -tim kilometrima ceste D2021 za prosti broj p .
- Svih k panoa treba biti postavljeno.

Primjer jednog ispravnog postavljanja $k = 4$ panoa na cestu D2021 duljine $n = 26$ km:



- Napišite rekurzivnu funkciju `br_nacina` (sami odredite argumente te funkcije), koja vraća broj načina na koje Pat i Mat mogu postaviti k panoa duž ceste D2021 duljine n kilometara uz poštivanje svih zahtjeva tvrtke za koju rade.
- Napišite program koji preko komandne linije prima dva prirodna broja n i k . Program treba, koristeći gornju funkciju, ispisati sve prirodne brojeve $i \leq k$ takve da je broj načina za postavljanje i panoa po gornjim pravilima prost broj. Ako komandna linija nema točno dva parametra, program ispisuje neku poruku o grešci, te vraća vrijednost 2021 operacijskom sustavu.

Programiranje 2 – popravni kolokvij, 3. 9. 2021.

Zadatak 2 (18+12 bodova) Chase i Ryder šalju si tajne poruke, pri čemu poruke šifriraju na ispod opisan način (opisani postupak možete pratiti na slici ispod).

- Ryder odabire prirodne brojeve n i k , pri čemu broj k dijeli broj n . Broj n predstavlja ukupan broj riječi u poruci. Pritom su poruke uvijek takve da je i svaka riječ poruke duljine najviše n . Također, riječi poruke sadrže samo mala slova engleske abecede (koja ima ukupno 26 slova).
- Riječi se redom upisuju na početke redaka matrice (ukoliko neka riječ nema n slova, nadopunjuje se slovima x).
- Dobivena $n \times n$ matrica dijeli se u $k \times k$ blokove. Znakovi svakog retka u svakom **bloku** pomiču se za 1 mjesto udesno (ako je to paran redak matrice) ili 1 mjesto ulijevo (ako je to neparan redak matrice - pritom retke brojimo od 0). Znak koji se nalazi na kraju retka bloka, ako je pomicanje udesno, vraća se na početak retka tog bloka, odnosno, ako je pomicanje ulijevo, znak s početka retka bloka stavlja se na kraj retka tog bloka.
- Mala slova zamjenjuju se svojim rednim brojem u engleskoj abecedi. Slovo 'a' zamijeni se brojem 1, 'b' brojem 2, itd.
- Brojevi koji se nalaze na glavnoj ili sporednoj dijagonali kvadriraju se.

Primjer za $n = 6$ i $k = 3$ (vrijedi da k dijeli n ✓):

$$\begin{pmatrix} k & o & l & | & a & c & i \\ n & a & x & | & x & x & x \\ a & k & c & | & i & j & i \\ \hline u & x & x & | & x & x & x \\ m & a & l & | & o & m & x \\ d & u & c & | & a & n & u \end{pmatrix} \rightarrow \begin{pmatrix} l & k & o & | & i & a & c \\ a & x & n & | & x & x & x \\ c & a & k & | & i & i & j \\ \hline x & x & u & | & x & x & x \\ l & m & a & | & x & o & m \\ u & c & d & | & n & u & a \end{pmatrix} \rightarrow \begin{pmatrix} 12 & 11 & 15 & 9 & 1 & 3 \\ 1 & 24 & 14 & 24 & 24 & 24 \\ 3 & 1 & 11 & 9 & 9 & 10 \\ 24 & 24 & 21 & 24 & 24 & 24 \\ 12 & 13 & 1 & 24 & 15 & 13 \\ 21 & 3 & 4 & 14 & 21 & 1 \end{pmatrix} \rightarrow \\ \rightarrow \begin{pmatrix} 144 & 11 & 15 & 9 & 1 & 9 \\ 1 & 576 & 14 & 24 & 576 & 24 \\ 3 & 1 & 121 & 81 & 9 & 10 \\ 24 & 24 & 441 & 576 & 24 & 24 \\ 12 & 169 & 1 & 24 & 225 & 13 \\ 441 & 3 & 4 & 14 & 21 & 1 \end{pmatrix}$$

- (a) Napišite funkciju `int** sifriraj(char** rijeci, int n, int k)` koja prima niz od n stringova i vraća (dinamički alociranu!) matricu `int`-ova dobivenu iz tih stringova na gore opisani način. Smijete koristiti dodatna (dinamički alocirana i ako treba dealocirana) polja, no ne smijete koristiti biblioteku `string.h` :)
- (b) U binarnoj datoteci '`sifrat.dat`' zapisan je cijeli broj n , te zatim $n \times n$ matrica cijelih brojeva (spremljena po **stupcima**). Napišite program koji otvara tu datoteku te provjerava je li matrica koja je u njoj zapisana nastala kao rezultat šifriranja neke poruke korištenjem gornjeg postupka. Ne treba odrediti početnu poruku, nego samo ispisati na ekran 'DA' ili 'NE', u ovisnosti o tome postoji li takva poruka. Također, nisu dozvoljena nikakva polja, te nije dozvoljeno pozivati funkciju iz (a) podzadatka.

Programiranje 2 – popravni kolokvij, 3. 9. 2021.

Zadatak 3 (25 bodova) Lolek i Bolek jako se vole igrati špekulama. Kad god bi skupili dovoljno novca, otišli bi u trgovinu i kupili jednu špekulu nekog proizvođača. Sve račune su vrijedno skupljali i otkrili da imaju 10000 špekula (računa). Lolek i Bolek žele dodijeliti vrijednost svakoj špekuli, prema broju špekula istog proizvođača koje posjeduju, na način da brojnije špekule vrijede manje a manje brojne špekule vrijede više. Lolek i Bolek su zaključili da mogu prikazati svoje račune kao vezanu listu cijelih brojeva, gdje svaki broj $k \in [0, 99]$ predstavlja jednu špekulu proizvođača k .

Pomognite Loleku i Boleku prebrojati koliko špekula kojeg proizvođača imaju tako da napravite funkciju koja će jednim prolazom velike liste računa (Lolek i Bolek nemaju puno vremena za pregledavanje računa) stvoriti listu parova (*proizvodac, brojSpekula*). Konačna lista parova mora biti sortirana silazno prema broju špekula određenog proizvođača. Poredak špekula proizvođača kojih imaju jednaki broj može biti proizvoljan. Pošto bi Lolek i Bolek što prije htjeli krenuti s igrom, sortiranje liste parova će vršiti učinkovitim merge sort algoritmom samo izmjenama veza elemenata (pokazivača).

Definirajte sve potrebne tipove za reprezentaciju početne i konačne liste, te napravite potrebne funkcije za dobivanje konačne sortirane liste parova iz koje će Lolek i Bolek jednostavno odrediti vrijednost za igru svake špekule koju posjeduju.

Napomene: Rješenje koje ne koristi merge sort s izmjenama pokazivača donosi maksimalno 10 bodova.

Programiranje 2 – popravni kolokvij, 3. 9. 2021.

Zadatak 4 (30 bodova) Jon želi nagovoriti Garfielda da prijede na vegetarijansku prehranu. Popis namirnica koje je kupio nalazi se u tekstualnoj datoteci `namirnice.txt`. Svaki njen redak sadrži *namirnicu* koja je ili (a) naziv, ili (b) pozitivan cijeli broj, malo slovo `x`, razmak te naziv (u slučaju (a), gdje je broj izostavljen, podrazumijeva se broj 1). Svaki naziv namirnice sadrži samo mala slova i razmake, ukupno najviše 49. Primjer sadržaja navedene datoteke je sljedeći:

```
4x sampinjon
umak od rajcice
brasno
2x maslac
sol
2x pasteta od slanutka
laneno ulje
muskatni orascic
1x kruh
3x smrznuta pizza
```

Dodatno, tekstualna datoteka `recepti.txt` sadrži barem jedan recept, i to jedan po retku, u sljedećem formatu: naslov recepta, dvotočka, razmak, pa jedna ili više namirnica odvojenih zarezom i razmakom. Svaki naslov recepta sadrži samo mala slova i razmake, ukupno najviše 49. Primjer:

```
lazanje sa sampinjonima: maslac, sol, brasno, muskatni orascic, umak od rajcice, 3x sampinjon
kruh i voda za dvije macke: 2x kruh, 2x voda
odmrznuta pizza: smrznuta pizza
```

Napišite program koji će u tekstualnu datoteku `preostaju.txt` (bilo kojim redom) zapisati točno one nazive recepata za jela koja je moguće pripremiti od kupljenih namirnica. Uz svaki recept treba napisati koliko će koje od namirnica u receptu preostati među kupljenima ako se recept napravi (s tim da broj 1 u izlaznoj datoteci ne smije biti implicitan kao u ulaznoj datoteci). U gornjem primjeru, ako su sve datoteke osim `namirnice.txt` i `recepti.txt` prazne, u `preostaju.txt` treba zapisati:

```
lazanje sa sampinjonima: 1x maslac, 1x sampinjon
odmrznuta pizza: 2x smrznuta pizza
```

Pritom nije potrebno provjeravati uspješnost otvaranja datoteka ni ispravnost njihovog sadržaja. **Nije dozvoljeno** deklarirati/alocirati/koristiti nizove osim najviše `tri` stringa pojedinačne duljine najviše 49 znakova niti koristiti zaglavlja osim `stdio.h` i `string.h`.

Programiranje 2 – popravni kolokvij, 3. 9. 2021.

Rezultati i uvidi: Rezultati u srijedu, 8.9.2021. navečer na webu. Informacije o uvidima bit će objavljene naknadno.

Upute: Na kolokviju je dozvoljeno koristiti samo pribor za pisanje i brisanje, te službeni podsjetnik. Kalkulatori, razne neslužbene tablice, papiri i sl., nisu dozvoljeni! **Mobitele isključite i spremite!** Sva rješenja napišite isključivo na papire sa zadacima, jer jedino njih predajete. Obavezno predajte sve papire sa zadacima, čak i ako neke zadatke niste rješavali. Ne zaboravite se **potpisati** na svim papirima! Skice smijete raditi i na drugim papirima koje će vam dati dežurni asistent/ica.

U svim zadacima **zabranjeno je korištenje dodatnih nizova** i standardne matematičke biblioteke (zaglavlje `math.h`), osim ako je u zadatku drugačije navedeno. Dozvoljeno je pisanje pomoćnih funkcija.

Uvjete navedene u zadacima (pozitivnost, ograde na n i sl.) ne treba provjeravati.

Zadatak 1 (20+5 bodova) Spužva Bob Skockani i Patrik Zvijezda zajedno su zaradili ukupno $k \in \mathbb{N}$ dolara. Srećom, održava se godišnji Rakburger Fest. Uz glavnu prometnicu, postavljeno je $n \in \mathbb{N}$ štandova, s rednim brojevima od 1 do n redom. Svaki štand prodaje svoju verziju rakburgera po cijeni od 9 dolara (po komadu). Bob i Patrick odlučili su svih k dolara potrošiti na te rakburgere (tj. kupovat će ih dok god imaju dovoljno dolara za barem jedan rakburger). Pritom se drže sljedećih pravila:

- štandove obilaze po redu, počevši od broja 1, pri čemu se nikad ne vraćaju unatrag (na štandove s manjim brojem);
 - na svakom štandu mogu kupiti više rakburgera (naravno ako imaju dovoljno dolara za platiti te rakburgere);
 - ne moraju na svakom štandu kupiti rakburgere, no moraju kupiti bar jedan rakburger na posljednja 2 štanda (ili posljednjem za $n = 1$);
 - zbog praznovjerja ne kupuju rakburgere na štandovima čiji broj p je prost;
 - na svakom sljedećem štandu na kojem kupe neke rakburgere, moraju kupiti strogo više rakburgera no što su ih kupili na prethodnom štandu (na kojem su kupovali rakburgere).
- (a) Napišite rekurzivnu funkciju `br_nacina` (sami odredite argumente te funkcije), koja vraća broj načina na koje Bob i Patrik mogu potrošiti k dolara duž niza od n štandova uz poštivanje svih zahtjeva navedenih u zadatku.
- (b) Napišite program koji preko komandne linije prima dva prirodna broja n i m . Program treba, koristeći gornju funkciju, ispisati sve prirodne brojeve $k \leq m$ takve da broj načina za potrošnju k dolara po gornjim pravilima **nije** niti prost broj niti 0. Ako komandna linija nema točno dva parametra, program ispisuje neku poruku o grešci, te vraća vrijednost 5 operacijskom sustavu.

Programiranje 2 – popravni kolokvij, 3. 9. 2021.

Zadatak 2 (18+12 bodova) Prilikom istraživanja, Dora je presrela poruku Lopuže Lisca koja je šifrirana. Šifrirana poruka zapravo je $n \times n$ matrica nenegativnih cijelih brojeva, a Dora je otkrila da tu poruku može dešifrirati na ispod opisan način (opisani postupak možete pratiti na slici ispod).

- Broj koji se nalazi na glavnoj ili sporednoj dijagonali uvijek je kvadrat nekog nenegativnog broja. Prema tome, Dora prvo mora odrediti korijen tih brojeva.
- U svakom retku nalazi se točno jedna nula. Sada, ako se radi o parnom retku (pri čemu retke brojimo od 0), treba elemente pomicati udesno sve dok 0 ne dođe na sporednu dijagonalu (pri čemu se element na kraju retka pomiče na početak tog retka). Ukoliko se radi o neparnom retku, elemente treba pomicati ulijevo sve dok 0 ne dođe na glavnu dijagonalu (pri čemu se element na početku retka pomiče na kraj tog retka; vidi na slici).
- Sada je svaki broj u matrici jedan od brojeva 0, 1, ..., 26. Svaki broj zamijenimo malim slovom, pri čemu broj i zamjenjujemo i -tim malim slovom u engleskoj abecedi: 1 zamijenimo s 'a', 2 s 'b', ..., 26 s 'z'. Broj 0 zamijenimo prazninom (tj. znakom ' ').
- Konačna poruka je niz od $n + 1$ stringova. Znakovi iz matrice prepisuju se redom u stringove - svaki string završava kad nađemo na znak za prazninu ' '. Pritom znakove čitamo redom po retcima matrice - iz parnih redaka slijeva nadesno, a iz neparnih zdesna nalijevo (vidi sliku!).

Primjer za $n = 6$:

$$\begin{pmatrix} 1 & 19 & 0 & 4 & 1 & 196 \\ 13 & 0 & 13 & 5 & 16 & 9 \\ 12 & 15 & 0 & 169 & 5 & 1 \\ 9 & 4 & 1 & 324 & 11 & 0 \\ 20 & 0 & 11 & 9 & 144 & 21 \\ 0 & 1 & 20 & 1 & 12 & 676 \end{pmatrix} \rightarrow \begin{pmatrix} 1 & 19 & 0 & 4 & 1 & 14 \\ 13 & 0 & 13 & 5 & 4 & 9 \\ 12 & 15 & 0 & 13 & 5 & 1 \\ 9 & 4 & 1 & 18 & 11 & 0 \\ 20 & 0 & 11 & 9 & 12 & 21 \\ 0 & 1 & 20 & 1 & 12 & 26 \end{pmatrix} \rightarrow \begin{pmatrix} 4 & 1 & 14 & 1 & 19 & 0 \\ 13 & 0 & 13 & 5 & 4 & 9 \\ 1 & 12 & 15 & 0 & 13 & 5 \\ 1 & 18 & 11 & 0 & 9 & 4 \\ 20 & 0 & 11 & 9 & 12 & 21 \\ 1 & 20 & 1 & 12 & 26 & 0 \end{pmatrix} \rightarrow \begin{pmatrix} d & a & n & a & s & \\ m & & m & e & d & i \\ a & l & o & & m & e \\ a & r & k & & i & d \\ t & & k & i & l & u \\ a & t & a & l & z & \end{pmatrix}$$

Poruka se sastoji redom od stringova: "danas", "idem", "malo", "medi", "krat", "kilu", "zlata".

- Napišite funkciju `char** desifriraj(int** sifrirano, int n)` koja prima prirodan broj $n > 2$ i $n \times n$ matricu cijelih brojeva `sifrirano`, te vraća (dinamički alocirano!) polje stringova koje predstavlja poruku dobivenu gore opisanim postupkom.
- U binarnoj datoteci `'sifrat.dat'` zapisan je cijeli broj n , te zatim $n \times n$ matrica cijelih brojeva (spremljena po **stupcima**). Napišite program koji otvara tu datoteku te provjerava je li matrica koja je u njoj zapisana nastala kao rezultat šifriranja neke poruke Lopuže Lisca korištenjem gornjeg postupka. Ne treba odrediti početnu poruku, nego samo ispisati na ekran 'DA' ili 'NE', u ovisnosti o tome postoji li takva poruka. Također, nisu dozvoljena nikakva polja, te nije dozvoljeno pozivati funkciju iz (a) podzadatka.

Programiranje 2 – popravni kolokvij, 3. 9. 2021.

Zadatak 3 (25 bodova) Lolek i Bolek su postavili čak 10000 balona kao ukras kod proslave rođendana njihovog prijatelja. Svaki balon je obojan jednom od 10 različitih boja. Zanima ih koliko balona pojedine boje su postavili. Tu informaciju planiraju saznati brojanjem balona redom kojim su ih postavljali, dakle baloni se mogu prikazati kao vezana lista cijelih brojeva, gdje svaki $k \in [0, 9]$ reprezentira jedan balon određene boje k .

Pomognite Loleku i Boleku prebrojati koliko balona koje boje su postavili tako da napravite funkciju koja će jednim prolazom velike liste balona (Lolek i Bolek žele što manje vremena potrošiti na brojanje balona) stvoriti listu parova (*boja, brojBalona*). Konačna lista parova mora biti sortirana uzlazno prema broju balona određene boje. Poredak balona onih boja koje su postavili jednaki broj može biti proizvoljan. Pošto bi se Lolek i Bolek što prije htjeli pridružiti proslavi, sortiranje liste parova će vršiti učinkovitim merge sort algoritmom samo izmjenama veza elemenata (pokazivača).

Definirajte sve potrebne tipove za reprezentaciju početne i konačne liste, te napravite potrebne funkcije za dobivanje konačne sortirane liste parova iz koje će Lolek i Bolek jednostavno saznati broj balona svake boje koje su postavili.

Napomene: Rješenje koje ne koristi merge sort s izmjenama pokazivača donosi maksimalno 10 bodova.

Programiranje 2 – popravni kolokvij, 3. 9. 2021.

Zadatak 4 (30 bodova) Jon želi nagovoriti Garfielda da prijede na vegetarijansku prehranu. Popis sastojaka koje je kupio nalazi se u tekstualnoj datoteci `sastojci.txt`. Svaki njen redak sadrži *sastojak* koji je ili (a) naziv, ili (b) pozitivan cijeli broj, malo slovo `x`, razmak te naziv (u slučaju (a), gdje je broj izostavljen, podrazumijeva se broj 1). Svaki naziv sastojka sadrži samo mala slova i razmake, ukupno najviše 49. Primjer sadržaja navedene datoteke je sljedeći:

```
muskatni orascic
maslac
sol
umak od rajcice
2x pasteta od slanutka
laneno ulje
brasno
4x sampinjon
1x kruh
```

Dodatno, tekstualna datoteka `recepti.txt` sadrži barem jedan recept, i to jedan po retku, u sljedećem formatu: naslov recepta, dvotočka, razmak, pa jedan ili više sastojaka odvojenih zarezom i razmakom. Svaki naslov recepta sadrži samo mala slova i razmake, ukupno najviše 49. Primjer:

```
lazanje sa sampinjonima: maslac, sol, brasno, muskatni orascic, umak od rajcice, 3x sampinjon
kruh i voda za dvije macke: 2x kruh, 2x voda
odmrznuta pizza: smrznuta pizza
```

Napišite program koji će u tekstualnu datoteku `treba_kupiti.txt` (bilo kojim redom) zapisati točno one nazive recepata za jela koja nije moguće pripremiti od kupljenih sastojaka. Uz svaki recept treba napisati koliko kojeg sastojka nedostaje (s tim da broj 1 u izlaznoj datoteci ne smije biti implicitan kao u ulaznoj datoteci). U gornjem primjeru, ako su sve datoteke osim `sastojci.txt` i `recepti.txt` prazne, u `treba_kupiti.txt` mora pisati:

```
kruh i voda za dvije macke: 1x kruh, 2x voda
odmrznuta pizza: 1x smrznuta pizza
```

Pritom nije potrebno provjeravati uspješnost otvaranja datoteka ni ispravnost njihovog sadržaja. **Nije dozvoljeno** deklarirati/alocirati/koristiti nizove osim najviše `tri` stringa pojedinačne duljine najviše 49 znakova niti koristiti zaglavlja osim `stdio.h` i `string.h`.