

Programiranje 1 – drugi kolokvij, 15. 02. 2021.

Upute: Na kolokvijju je dozvoljeno koristiti samo pribor za pisanje i brisanje. Kalkulatori, razne neslužbene tablice, papiri i sl., nisu dozvoljeni! Mobitele isključite i spremite! 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.

Napomena: Svi zadaci su programski zadaci, u smislu uvjeta polaganja kolegija (80% bodova na barem jednom zadatku). 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.

Zadatak 1 (15=8+6+1 bodova)

- (a) Napišite funkciju `znamenke_fja` koja prima prirodne brojeve n i b , pri čemu vrijedi $b \geq 2$. Funkcija treba vratiti najveću znamenku u zapisu broja n u bazi b , te preko varijabilnih argumenata sumu svih neparnih znamenaka i sumu svih parnih znamenaka.

Primjerice, za $n = 2020$ i $b = 6$, vrijedi $2020 = (13204)_6$, pa je najveća znamenka u tom prikazu 4, dok je suma neparnih znamenki $3 + 1 = 4$, a suma parnih znamenki $4 + 0 + 2 = 6$. (Podsjetnik: prazna suma iznosi 0.)

- (b) Napišite funkciju `znamenke_razlika` koja prima prirodne brojeve m i n , pri čemu je $m \leq n$. Funkcija vraća prirodan broj k , $m \leq k \leq n$, koji ima najveću razliku sume parnih znamenaka i sume neparnih znamenaka u bazi 5. Ukoliko je više takvih brojeva, vratite najveći od njih. Smijete koristiti funkciju iz (a) podzadatka (čak i ako ju niste napisali).

Primjerice, za $m = 2020$ i $n = 2022$, imamo redom prikaze prirodnih brojeva k , $2020 \leq k \leq 2022$, u bazi 5: $2020 = (31040)_5$, $2021 = (31041)_5$, $2022 = (31042)_5$. Za te brojeve, tražene razlike suma parnih i suma neparnih znamenaka iznose redom: 0, -1, 2, iz čega slijedi da je traženi broj $k = 2022$.

- (c) Napišite (cijelu) funkciju `int main(void)` u kojoj na konkretnim primjerima ilustrirate pozive funkcija iz (a) i (b) podzadatka (čak i ako te funkcije niste napisali!).

Programiranje 1 – drugi kolokvij, 15. 02. 2021.

Zadatak 2 (20 bodova)

- (a) Napišite funkciju `polinom` koja prima cijeli broj a , prirodan broj n i realan broj x te Hornerovim algoritmom računa i vraća vrijednost izraza

$$p(x) = \sum_{i=0}^n a_i x^i$$

pri čemu je

$$a_i = \begin{cases} \binom{i}{|a|}, & \text{ako je } |a| \leq i \\ \binom{|a|}{i}, & \text{ako je } |a| > i \end{cases}$$

- (b) Napišite funkciju `main` u kojoj učitava prirodan broj n , polje cijelih brojeva b s n elemenata, te učitani niz sortira **silazno**, po vrijednosti `polinom(b[i], n, x)` pri čemu je x aritmetička sredina unesenih elemenata polja. Možete pretpostaviti da polje nema više od 100 elemenata. Također, smijete koristiti funkciju iz (a) zadatka bez da ste ga riješili.

Programiranje 1 – drugi kolokvij, 15. 02. 2021.

Zadatak 3 (15 bodova) Za različite brojeve $m, n \in \mathbb{N}$ kažemo da su *prijateljski* ako je zbroj pravih djelitelja od m (*pravi* ovdje znači *različiti od m*) jednak n te je zbroj pravih djelitelja od n (opet, različitih od n) jednak m . Najmanji takvi brojevi su 220 i 284. Naime, zbroj pravih djelitelja od 220 je

$$1 + 2 + 4 + 5 + 10 + 11 + 20 + 22 + 44 + 55 + 110 = 284,$$

a zbroj pravih djelitelja od 284 je

$$1 + 2 + 4 + 71 + 142 = 220.$$

Napišite funkciju `del_amicable` koja prima niz prirodnih brojeva (kao i obično, s njegovom duljinom) i iz njega briše jedan par prijateljskih brojeva (bilo koji ako postoji; inače ispisuje odgovarajuću poruku). Povratna vrijednost funkcije je novi broj elemenata niza.

Primjer: Funkcija `del_amicable` će niz

1 2 3 220 4 220 5 284 6 284

duljine 10 promijeniti, recimo, u sljedeći niz duljine 8 (i vratiti 8):

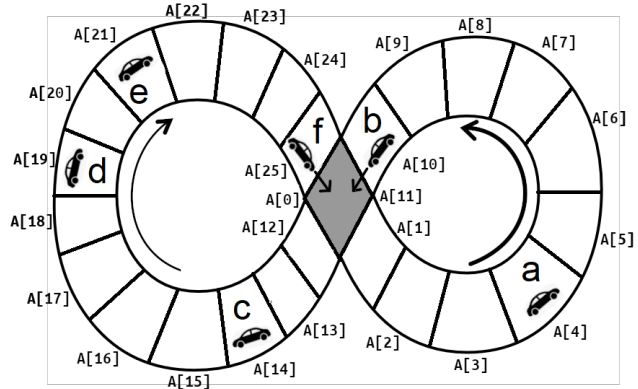
1 2 3 4 220 5 6 284.

Napomena: Dodatne (pomoćne) funkcije su poželjne, ali dodatni nizovi nisu dozvoljeni.

Programiranje 1 – drugi kolokvij, 15. 02. 2021.

Zadatak 4 (20 bodova)

Dani su prirodni brojevi $k < n$. Kružna staza je dana poljem A elemenata tipa char duljine n . Auti na njoj dani su različitim slovima male engleske abecede, ostala mjesta označena su s '+'. U svakoj sekundi svaki auto na stazi na mjestu i pomiče se na mjesto s indeksom $i+1$ (posebno, auto na mjestu s indeksom $n-1$ pomiče se na mjesto s indeksom 0). Staza na indeksu k siječe samu sebe na mjestu s indeksom 0, stoga auti koji se u nađu na tim mjestima u istom trenu sudaraju se i diskvalificiraju.



Primjerice, za parametre $n = 25$, $k = 11$ prikazana je jedna moguća situacija na stazi i odgovarajuće stanje polja A .

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25
+	+	+	+	a	+	+	+	+	+	b	+	+	+	c	+	+	+	+	d	+	e	+	+	+	f

- a) Napišite funkciju `int PrviSudar (char A[], int n, int k, char* auto1, char* auto2)` koja simulira utrku i nalazi prvi trenutak kada se neka dva auta sudare. Funkcija vraća broj sekundi potrebnih dođe do sudara (ili -1 , ako nije bilo sudara), a preko varijabilnih parametara oznake sudarenih auta. Nakon izvođenja funkcije, u polju A nalazi se stanje nakon sudara (ili isto stanje kao prilikom poziva funkcije, ako nije bilo sudara).

Pozvana na stanje sa slike gore, funkcija vraća 1, aute b i f , a stanje u polju A je dano na sljedeći način:

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25
+	+	+	+	a	+	+	+	+	+	+	+	+	+	c	+	+	+	+	d	+	e	+	+	+	

- b) Napišite funkciju `int JediniPrezivjeli (char A[], int n, int k, char* auto1)` koja vraća 1 ako postoji auto koji će u nekom trenutku ostati sam na stazi (0, ako ne postoji), te preko varijabilnog argumenta njegovu oznaku. Na kraju izvođenja funkcije u polju A mora biti stanje nakon zadnjeg sudara (ili isto stanje kao prilikom poziva funkcije, ako nije bilo sudara).

U rješenju nije dozvoljeno deklarirati, alocirati ni koristiti dodatna polja!