

Tekstovi zadataka

1. Pravi, netrivialni delioci celog broja su svi delioci osim jedinice i samog tog broja (trivialni delioci su 1 i sam taj broj). Napisati program koji za uneti pozitivan ceo broj n ispisuje sve njegove prave delioce. U slučaju neispravnog unosa, ispisati odgovarajuću poruku o grešci i prekinuti program.
2. Dopuniti program iz 1.zadatka tako da se u slučaju kada uneti broj nema nijednog pravog delioca ispiše odgovarajuću poruku. Zadatak uraditi na dva načina:
 - 1) pomoću brojačke promenljive
 - 2) pomoću indikatorske promenljive
3. Napisati program koji za uneti pozitivan ceo broj n proverava da li je prost. Broj je prost ukoliko je deljiv samo 1 i samim sobom, odnosno ako nema netrivialne delioce. Podrazumevati da je ulaz ispravan, tj. da je $n \geq 0$. Program treba u svakom od slučajeva da ispiše odgovarajuću poruku na standardni izlaz. Zadatak uraditi na dva načina:
 - 1) pomoću brojačke promenljive
 - 2) pomoću indikatorske promenljive
4. Napisati program koji ispituje da li je dati broj savršen. Broj je savršen ako je jednak zbiru svih svojih pravih (netrivialnih) delilaca, uključujući i 1. Podrazumevati da je ulaz ispravan, tj. da je $n \geq 0$. Program treba u svakom od slučajeva da ispiše odgovarajuću poruku na standardni izlaz.
5. Napisati program koji učitava nenegativan ceo broj i ispisuje njegove cifre u obrnutom poretku. U slučaju neispravnog unosa, ispisati odgovarajuću poruku o grešci i prekinuti program.
6. Napisati program koji za uneti nenegativan ceo broj ispisuje koliko ima cifara u dekadnom zapisu. U slučaju neispravnog unosa, ispisati odgovarajuću poruku o grešci i prekinuti program. Nije dozvoljeno koristiti karakterski pristup i ugrađenu funkciju `len`.
7. Napisati program koji za uneti pozitivan ceo broj n proverava i ispisuje da li se cifra 5 nalazi u njegovom zapisu. Podrazumevati da je ulaz ispravan, tj. da je $n \geq 0$. Program treba u svakom od slučajeva da ispiše odgovarajuću poruku na standardni izlaz.
8. Napisati program koji za uneti pozitivan ceo broj ispisuje da li je on deljiv sumom svojih cifara. U slučaju neispravnog unosa, ispisati odgovarajuću poruku o grešci i prekinuti program. Program treba u svakom od slučajeva da ispiše odgovarajuću poruku na standardni izlaz.
9. Napisati program koji sa standardnog ulaza učitava jedan string i ispisuje njegove karaktere pojedinačno. Zadatak uraditi na tri načina:
 - 1) pomoću `while` petlje iteriranjem po indeksima karaktera
 - 2) pomoću `for` petlje iteriranjem po indeksima karaktera
 - 3) pomoću `for` petlje iteriranjem po karakterima
10. Napisati program koji sa standardnog ulaza učitava jedan string, a potom računa i ispisuje broj malih slova, broj velikih slova, broj cifara i broj belina sadržanih u datom stringu.
11. Napisati program koji učitava stringove sve do unosa stringa **"KRAJ"** kao oznake za kraj unosa, računa i ispisuje ukupan broj unetih samoglasnika i ukupan broj unetih suglasnika. Ne praviti razliku između malih i velikih slova.
12. Modifikovati program iz 11.zadatka tako da se stringovi učitavaju sve do kraja ulaza, tj. do unosa `EOF` indikatora za kraj ulaza.
13. Napisati program koji sa standardnog ulaza učitava jedan string i na osnovu njega pravi novi, tzv. *konvertovani* string tako što mala slova zamenjuje odgovarajućim velikim slovima, a velika slova zamenjuje odgovarajućim malim slovima, dok svi ostali karakteri ostaju neizmenjeni. Program ispisuje konvertovani sting na standardni izlaz.
14. String je *palindrom* ukoliko se isto čita i sa leve i sa desne strane. Napisati program koji proverava da li je učitani string palindrom. Program treba u svakom od slučajeva da ispiše odgovarajuću poruku na standardni izlaz.

15. Napisati program koji za učitani string **s** i karakter **c** (string koji se sastoji samo od jednog karaktera) ispituje da li se karakter **c** pojavljuje u stringu **s**. Program treba u svakom od slučajeva da ispiše odgovarajuću poruku na standardni izlaz. Nije dozvoljeno koristiti operator **in** kao ni metode **find** ili **index** klase **str**.
16. Dopuniti program iz 15. zadatka tako da se u slučaju string **s** sadrži karakter **c** ispisuje i indeks prvog pojavljivanja karaktera **c** u stringu **s**.
17. Napisati program koji za dva učitana stringa **s** i **t** proverava da li je string **t** podstring stringa **s**. Program treba u svakom od slučajeva da ispiše odgovarajuću poruku na standardni izlaz. Podrazumevati da je ulaz ispravan, tj. da je string **t** manje ili jednake dužine od stringa **s**. Nije dozvoljeno koristiti operator **in** kao ni metode **find** ili **index** klase **str**.
18. Napisati program koji za unetu reč (string) i pozitivan ceo broj **n** formira rezultujuću reč tako što unetu reč kopira **n** puta pri čemu se između svaka dva kopiranja umeće crtica. U slučaju neispravnog unosa, ispisati odgovarajuću poruku o grešci i prekinuti program.
19. Napisati program koji obrće učitani string **s** i ispisuje na standardni izlaz obrnuti string. Nije dozvoljeno koristiti slajsing operator.
20. Napisati program koja ciklično rotira učitani string **s** za **k** mesta udesno. Program učitava string **s** i nenegativan ceo broj **k** i ispisuje rotirani string. U slučaju neispravnog unosa, ispisati odgovarajuću poruku o grešci i prekinuti program. Nije dozvoljeno koristiti slajsing operator.
21. Napisati program koji vrši leksikografsko poređenje dva stringa. Program učitava dva stringa **s** i **t** i ispisuje odgovarajuću poruku na standardni izlaz. Nije dozvoljeno korišćenje operatora **<** nad celim stringovima.

Zadaci za vežbanje

22. [Žmurke] Napisati program koji simulira brojanje od 0 do 100 kao kod dečije igre "žmurke", gde igrač koji trazi preostale igrače broji redom 0, 5, 10, 15, 20, ..., 100. Nakon odbrojanja program ispisuje poruku: "Krecem potragu!".
23. [StudAssistApp] Napisati program gde korisnik unosi prvo broj indeksa studenta i broj položenih ispita, a zatim i ocene sa svih položenih ispita (ocene su brojevi 6-10). Nakon toga aplikacija računa i na standardni izlaz ispisuje studentov prosek ocena. U slučaju greške ispisati odgovarajuću poruku i prekinuti program.
24. Modifikovati program iz prethodnog zadatka tako da se u slučaju nevalidne ocene ne prekida program, već vraća korisnika da ponovo unese tu ocenu.
25. [Tajni broj] Napisati igricu pogađanja tajnog broja. Program slučajnim izborom bira ceo broj iz intervala [0, 100]. Nakon toga, igrač unosi korisničko ime i kreće sa pogađanjem. U svakom pokušaju, program igraču daje povratnu informaciju da li je uneti broj je manji ili veći od tajnog broja ili poruku o uspesnom pogađanju.
26. Modifikovati program iz prethodnog zadatka tako da se igraču računa score (broj pokušaja). Prilikom uspešnog pogađanja broja ispisati i vrednost skora.
27. [BankAssist] Napisati program koji simulira rad banke. Program učitava pocetnu kolicinu novca (u EUR), zatim procenat godisnje kamate zatim broj godina. Nakon toga, program racuna i na standardni izlaz ispisuje kolika ce suma novca biti nakon zadatih godina. Pretpostaviti da ce ulazni podaci biti validni.

PRIMER: pocetni kapital 10 000EUR, kamatna stopa 3%, broj godina 3

nakon 1.godine: $10000\text{EUR} * 1.03 = 10300\text{EUR}$

nakon 2.godine: $10300\text{EUR} * 1.03 = 10609\text{EUR}$

nakon 3.godine: $10609\text{EUR} * 1.03 = 10927.27\text{EUR}$