Stringovi

1. Napisati funkciju String velikaSlova(String s) koja transformiše string s tako što između svaka dva uzastopna velika slova umetne karakter ‘\_’. Na primjer, unijeti string s = "aBCDeF3GH", funkcija treba da transformiše u "aB\_C\_DeF3G\_H".
2. Napisati funkciju String umetniSlovo(String s) koja transformiše string s tako što nakon svakog velikog slova u stringu umetne njemu odgovarajuće malo slovo. Na primjer, unijeti string S="ProGraMiraNje", funkcija treba da transformiše u "PproGgraMmiraNnje".
3. Napisati funkciju String samoglasnici(String s) koja transformiše string s tako što između svaka dva uzastopna samoglasnika umetne karakter ‘-’. Na primjer, unijeti string s = "AbcEaoU7", funkcija treba da transformiše u "AbcE-a-o-U7".
4. Napisati funkciju String cifre(String s) koja transformiše string s tako što nakon svake parne cifre u stringu umetne slovo ‘P’, a nakon svake neparne cifre slovo ‘N’. Na primjer, unijeti string S="ab2cg3#1", funkcija treba da transformiše u "ab2Pcg3N#1N".
5. Napisati funkciju String umetniCifru(String s) koja transformiše taj string s tako što nakon svake cifre veće od 0 u stringu umetne cifru za jedan manju, a nakon cifre 0 umetne cifru 9. Na primjer, unijeti string s = "ab2c9#03", funkcija treba da transformiše u "ab21c98#0932".
6. Napisati funkciju String premjesti(string s) koja od stringa s pravi oni string u kojem su samoglasnici ispred suglasnika. Možete pretpostaviti da će se s sastojati samo od malih slova.
7. Napisati program koji učitava dva stringa jednake dužine i od njih pravi treći string tako što u njemu naizmjenično postavlja slova iz jednog, pa drugog stringa. Na primjer, za stringove “pmf“ i “abc“ rezultat treba da bude “pambfc“.
8. Napisati program koji provjerava validnost unijete šifre. Šifra se smatra validnom ukoliko ima barem 8 karaktera i dodatno ispunjava sljedeće kriterijume
   * Sadrži barem jedno veliko i barem jedno malo slovo abecede,
   * Sadrži barem jednu cifru,
   * Sadrži barem jedan od specijalnih karaktera iz skupa {$,#,@}.
9. Napisati funkciju String saberi(String a, String b) koja sabira binarne brojeve a i b koji su zadati kao stringovi. Na primjer, ako je a=“111“ i b = “11”, rješenje je “1000”.
10. Rimske cifre predstavljene su simbolima I=1, V=5, X=10, L=50, C=100, D=500, M=1000. Na primjer, broj 7 se rimskim ciframa piše kao “VII” (5+1+1), 63 kao “LXIII” (50+10+1+1+1), 2021 kao “MMXXI” (1000+1000+10+10+1) itd. Rimski brojevi pišu se od većih cifara ka manjim, osim u nekoliko posebnih slučajeva:
    * I se može naći ispred V i X kako bi se dobili brojevi 4 i 9.
    * X se može naći ispred L i C kako bi se dobili brojevi 40 i 90.
    * C se može naći ispred D i M kako bi se dobili brojevi 400 i 900.

Napisati program koji sa standardnog ulaza učitava string koji predstavlja rimski broj i izračunava i štampa njegovu vrijednost u dekadnom brojnom sistemu.

1. Rimske cifre predstavljene su simbolima I=1, V=5, X=10, L=50, C=100, D=500, M=1000. Na primjer, broj 7 se rimskim ciframa piše kao “VII” (5+1+1), 63 kao “LXIII” (50+10+1+1+1), 2021 kao “MMXXI” (1000+1000+10+10+1) itd. Rimski brojevi pišu se od većih cifara ka manjim, osim u nekoliko posebnih slučajeva:
   * I se može naći ispred V i X kako bi se dobili brojevi 4 i 9.
   * X se može naći ispred L i C kako bi se dobili brojevi 40 i 90.
   * C se može naći ispred D i M kako bi se dobili brojevi 400 i 900.

Napisati program koji sa standardnog ulaza prirodan broj N < 4000 i koji štampa odgovarajući broj zapisan rimskim ciframa.

1. Napisati program koji sa standardnog ulaza učitava string koji predstavlja evidenciju o prisustvovanju studenta predavanjima. Ovaj string sastoji se iz karaktera P, O, Z. Karakter P označava da je student prisustvovao času, karakter O da je student odsustvovao sa časa, a karakter Z da je student zakasnio na čas. Program treba da provjeri da li će student dobiti poene za prisustvo i odštampa odgovarajuću poruku. Student će dobiti poene za prisustvo ukoliko nije odsustvovao sa više od dva časa uzastopno i ako nije zakasnio na više od tri uzastopna časa. Na primjer, ako je zapis prisustva “PPOOLLLOP”, student će dobiti poene za prisustvo, dok student sa sljedećim zapisom “PPOOLOOOP” neće dobiti poene za prisustvo.
2. Napisati program koji provjerava da li je unijeti string oblika 𝑎2n𝑏m𝑐2n za proizvoljne prirodne brojeve m i n. Na primjer, aabcc i aaaabbcccc jesu riječi ovog oblika, dok aaabccc nije.
3. Napisati program koji za data dva stringa S i T provjerava da li su jednaki kada se otkucaju na tastaturi. Karakter ‘#’ ima specijalno značenje i predstavlja pritisak backspace dugmeta na tastaturi (brisanje karaktera ispred kursora).

Ulaz: U prvom redu ulaza dat je string S. U drugom redu ulaza dat je string T.

Izlaz: U jedinom redu izlaza odštampati “DA” ukoliko su dva stringa jednaki, a “NE” u suprotnom. Primjeri:

|  |  |  |
| --- | --- | --- |
| Ulaz: | Ulaz: | Ulaz: |
| ab#c  ad#c | ##ab#  c#d#a | b#c  #bc |
| Izlaz: | Izlaz: | Izlaz: |
| DA | DA | NE |

1. Napisati funkciju String cezar(string s, int k) koja šifrira Cezarovom šifrom string s koji sadrži isključivo mala slova engleske abecede. Cezarovo šifriranje se vrši tako što se svako slovo ciklično pomjeri za k mjesta udesno po abecedi. Na primjer za k=2 ’a’ se šifrira kao ’c’, ’b’ kao ’d’ itd.
2. Viženerov algoritam za enkripciju stringova funkcioniše na sljedeći način. Ključ enkripcije je neki string proizvoljne dužine, na primjer “test”. Šifrirani string se dobija u dva koraka. Prvo se od datog ključa formira prošireni ključ čija je dužina jednaka dužini ulaznog stringa. Ovo se postiže tako što se ključ ponavlja dok ne dostigne potrebnu dužinu. Na primjer, za ulazni string “abcdefwxyz” prošireni ključ bi bio “testtestte”. Potom se karakteri na istim indeksima u ulaznom stringu i proširenom ključu sabiraju po modulu 26. U prethodnom primjeru bi šifrirani string bio “tfuwxjoqrd”. Dekripcija je postupak kojim se od enkriptovanog stringa, uz pomoć ključa, dobija originalni string. Ona se vrši oduzimanjem karaktera proširenog ključa od karaktera šifriranog stringa po modulu 26. Napisati funkcije String enkripcija(String tekst, String kljuc) i String dekripcija(String tekst, String kljuc) koje kao argumente primaju ulazni string i ključ, i vrše enkripciju i dekripciju ulaznog stringa, respektivno. Test primjeri će se sastojati isključivo od malih slova engleskog alfabeta.
3. Napisati program koji za učitani string nalazi i štampa najduži podstring datog stringa koji je palindrom. Na primjer za string: “babamamabbaaab” odgovor je: “bamamab”.
4. Napisati program koji sa standardnog ulaza učitava string genom koji se sastoji od velikih slova ’A’, ’C’, ’G’, ’T’ (skraćenice za 4 nukleotidne baze adenin, citozin, guanin i timin) i broj k. Vaš program treba da pronađe i odštampa sve podstringove stringa genom dužine k koji sadrže sva četiri slova. Ne štampati duplikate.

|  |  |
| --- | --- |
| Ulaz: | Ulaz: |
| AAAACTGA  4 | AAAAACCCCCAAAAAGGGTTTCCCAAAAAGGGTTT  10 |
| Izlaz: | Izlaz: |
| ACTG  CTGA | CAAAAAGGGT  GTTTCCCAAA  TCCCAAAAAG |

1. Napisati program koji sa standardnog ulaza učitava string genom koji se sastoji od velikih slova ’A’, ’C’, ’G’, ’T’ (skraćenice za 4 nukleotidne baze adenin, citozin, guanin i timin) i prirodan broj k. Vaš program treba da pronađe i odštampa sve podstringove stringa genom dužine k koji se u njemu javljaju više od jednom. Ne štampati duplikate.

|  |  |
| --- | --- |
| Ulaz: | Ulaz: |
| AAAAAAAAAAAAA  5 | AAAAACCCCCAAAAACCCCCCAAAAAGGGTTT  10 |
| Izlaz: | Izlaz: |
| AAAAA | AAAAACCCCC  AAAACCCCCA  AAACCCCCAA  AACCCCCAAA  ACCCCCAAAA  CCCCCAAAAA |

1. Napisati funkciju bool anagram(string str1, string str2) kojom se provjerava da li su dva data stringa anagrami. Stringovi su anagrami ukoliko se jedan string može dobiti premještanjem slova drugog stringa. Na primjer stringovi “ortoped“ i “torpedo“ su anagrami.
2. Napisati program koji učitava dvije niske karaktera koje predstavljaju elemente dva skupa. Skupovi nemaju više od 26 elemenata. Program treba da odredi i odštampa niske koje predstavljaju uniju, presjek i simetričnu razliku dva skupa. Elementi niski su mala slova engleske abecede.
3. Napisati program koji sa standardnog ulaza učitava dva stringa A i B i koji izračunava koliko najviše puta se neki od karaktera iz stringa A nađe u stringu B.

Primjeri:

|  |  |
| --- | --- |
| Ulaz: | Ulaz: |
| aA  aAAbbbb | abc  ABCdef |
| Izlaz: | Izlaz: |
| 3 | 0 |

1. Napisati program koji sa standardnog ulaza učitava dva stringa A i B koji sadrže isključivo mala slova engleske abecede. Program treba da pronađe i odštampa slova po kojima se stringovi A i B razlikuju (da jedan sadrži određeno slovo a drugi ne). Svako ponavljanje određenog slova, posmatra se odvojeno.

Primjeri:

|  |  |
| --- | --- |
| Ulaz: | Ulaz: |
| bcae  cdba | abaad  baab |
| Izlaz: | Izlaz: |
| ed | adb |

Objašnjenje: Za stringove "abaad" i "baab" štampa se "adb" jer string A sadrži tri slova 'a', a string B dva. String B sadrži dva slova 'b', dok string A sadrži jedno. Slovo 'd' pojavljuje se samo u stringu A.

1. Napisati program koji sa standardnog ulaza učitava dva stringa A i B i provjerava da li se string A može sastaviti korišćenjem isključivo slova iz stringa B. Ukoliko je moguće odštampati „DA“. U suprotnom, odštampati „NE“.

Primjeri:

|  |  |  |
| --- | --- | --- |
| Ulaz: | Ulaz: | Ulaz: |
| a  b | aa  ab | aa  aba |
| Izlaz: | Izlaz: | Izlaz: |
| NE | NE | DA |

1. IP adresa je jedinstvena oznaka hosta u računarskoj mreži. IP adresa se sastoji od četiri prirodna broja od 0 do 255 (bez vodećih nula) odvojena tačkama. Napisati program koji sa standardnog ulaza učitava string i provjerava da li taj string predstavlja validnu IP adresu. Primjeri validnih IP adresa su: “192.168.1.0”, “255.255.0.0”, “0.1.2.201”. Primjeri nevalidnih IP adresa su: “2.011.255.245”, “192.168.1.312”, “192.168.1x1”.
2. IP adresa je jedinstvena oznaka hosta u računarskoj mreži. IP adresa se sastoji od četiri prirodna broja od 0 do 255 (bez vodećih nula) odvojena tačkama. Primjeri validnih IP adresa su: “192.168.1.0”, “255.255.0.0”, “0.1.2.201”. Primjeri nevalidnih IP adresa su: “2.011.255.245”, “192.168.1.312”, “192.168.11”.

Napisati program koji sa standardnog ulaza učitava string koji se sastoji samo od cifara i koji računa koliko se validnih IP adresa može kreirati dodavanjem tačaka u taj string. Na primjer, ako je s=”25525511135”, odgovor je 2 (mogu se formirati IP adrese “255.255.111.35” i “255.255.11.135”). Ako je s=”0000”, odgovor je 1 (adresa “0.0.0.0”). Ako je s=”101023”, odgovor je 5 (“1.0.10.23”, “1.0.102.3”, “10.1.0.23”, “10.10.2.3” i “101.0.2.3”).

1. Napisati funkciju String zamijeni(String s, String str1, String str2) koja vraća string dobijen zamjenom svakog pojavljivanja stringa str1 u stringu s, stringom str2.
2. Napisati program koji okreće redosljed karaktera u riječima unutar unijetog stringa. Na primjer, ako je dat string „Kurs iz programskog jezika Java“, rezultat je: „surK zi goksmargorp akizej avaJ“.
3. Napisati funkciju bool validanEmail(String tekst) koja provjerava da li dati tekst predstavlja validnu email adresu. Validna email adresa sastoji se od email prefiksa i email domena između kojih je symbol “@”. Email prefiks može sadržati mala slova engleske abecede, brojeve i karaktere “\_”, “-” i “.”. Prefiks mora sadržati barem jedno slovo ili broj i mora počinjati i završavati se slovom ili brojem. Domen se sastoji od malih slova, brojeva i karaktera “-” i “.”. Poslednja dva karaktera domena moraju biti slova. Ni prefiks ni domen ne smiju sadržati dvije uzastopne tačke.
4. Napisati program koji učitava datum u jednom od sljedećih formata MM-DD-YYYY, MM/DD/YYYY ili MM.DD.YYYY i štampa datum narednog dana. Na primjer, ako je ulaz 31.12.2020, program treba da odštampa 01.01.2021.
5. Napisati funkciju bool validanDatum(String tekst) koja provjerava da li dati tekst predstavlja validan datum. Datum sadrži informacije o danu, mjesecu i godini u nekom od sljedećih formata:
   * DD.MM.YYYY – po tačno dvije cifre za dan i mjesec i četiri cifre za godinu.

Primjer: 21.06.2021

* D.M.YYYY – dan i mjesec bez vodećih nula i godina sa četiri cifre.

Primjer: 21.6.2021

* D. Month YYYYY – dan bez vodećih nula, mjesec sa punim imenom i godina sa četiri cifre.

Primjer: 1. januar 2021

* D. Mon YYYY – dan, mjesec sa skraćenim zapisom od tri karaktera i četiri cifre za godinu.

Primjer: 1. jan 2021

Kod prva dva formata, umjesto separatora “.” mogu se koristiti i karakteri “/” i “-”, ali se

u jednom zapisu validnog datuma može nalaziti samo jedan od njih.