

I smer, Programiranje 2 2016/2017, ispit, JUN 2

Na *Desktop*-u napraviti direktorijum čije je ime u formatu **InicijaliAsistenta_ImeIPrezime_BrojIndeksa**. Na primer, **AZ_PeraPeric_mi15231**. Sve zadatke sačuvati u ovom direktorijumu. Zadatke imenovati sa **1.c**, **2.c**, **3.c**, **4.c**, **5.c**. Za kompilaciju 4. i 5. zadatka koristiti naredbe **gcc 4.c liste.c i gcc 5.c stabla.c**

1. Kao argumenti komandne linije se redom zadaju pozitivan ceo broj p i niska s . Napisati program koji u datoteku **podniske.txt** ispisuje redom sa leva na desno sve podniske od s , dužine p . Na primer, za $p = 3$ i nisku "program", potrebno je ispisati "pro", "rog", "ogr", "gra", "ram". U slučaju greške ispisati -1 na standardni izlaz za greške.

Pozivanje	./a.out 3 drvo	./a.out 1 MATF	./a.out 2 racunar	./a.out 5	./a.out 15 ispit	./a.out -2 oblak
podniske.txt	drv rvo	M A T F	ra ac cu un na ar			
Izlaz za greške				-1		-1

2. U datoteci **ispiti.txt** se nalaze informacije o terminima ispitima. Za svaki ispit su poznati: šifra predmeta (niska maksimalne dužine 15 karaktera), dan, mesec i godina održavanja ispita (pozitivni celi brojevi). Nije unapred poznat broj podataka u datoteci. Napisati program koji čita informacije iz navedene datoteke, a zatim sortira podatke po datumu rastuće. Ako su dva ispita raspoređena da budu istog dana (ako im se poklapaju i dan i mesec i godina), onda ih treba sortirati po šifri leksikografski opadajuće. Na standardni izlaz ispisati šifre predmeta iz sortiranih niza ispita. U slučaju greške, ispisati -1 na standardni izlaz za greške.

ispiti.txt	P2 1 7 2017 VEB 4 7 2016 P1 22 6 2015	P1 21 12 2012 P2 21 12 2012 VEB 21 12 2012	P1 21 09 2016 P2 15 09 2016 RBP 30 10 2015	<i>prazna datoteka</i>	P1 1 1 2001 -15
Izlaz	P1 VEB P2	VEB P2 P1	RBP P2 P1		
Izlaz za greške					-1

3. Napisati funkciju koja korišćenjem bitovskih operatora menja neoznačen ceo broj na sledeći način:

- Ako se u bitovskoj reprezentaciji broja na i -toj poziciji nalazi nula između dve jedinice, 0 na i -toj poziciji postaje 1.
- Ako se u bitovskoj reprezentaciji broja na i -toj poziciji nalazi jedinica između dve nule, 1 na i -toj poziciji postaje 0.

Svi ostali bitovi ostaju nepromenjeni. Bitove menjati sa desna na levo. Ako se posmatra i -ti bit, u obzir se uzima izmenjeni $(i - 1)$ -vi bit i vec postojeći $(i + 1)$ -vi bit. Sa standardnog ulaza se učitava ceo neoznačen broj x . Na standardni izlaz ispisati broj koji se dobija izvršavanjem napisane funkcije za broj x .

Primer:

721233 = 00000000000010110000000101010001

-> 00000000000011110000000000000001 = 983041

Ulaz	0	1431655765	2863311530	42
Izlaz	0	2147483647	2147483648	0

4. Napisati funkciju koja iz liste, čiji su elementi celi brojevi, briše sve elemente čija je vrednost veća od zbira svih elemenata koji se u listi nalaze posle njega. Sa standardnog ulaza se unose elementi liste sve do kraja ulaza. Na standardni izlaz ispisati rezultujuću listu.

Ulaz	1 2 3 4 5	20 9 4 2 1 0	5 -1 -2 -3 -1 -1	10 1 7 2 3 0	<i>prazna lista</i>
Izlaz	[1, 2, 3, 4, 5]	[0]	[-1, -1]	[10, 1, 2, 0]	[]

5. Iz datoteke **stablo.txt** se učitavaju celobrojne vrednosti čvorova stabla. Napisati funkciju koja računa proizvod svih čvorova stabla za koje važi da se nalaze na parnim nivoima stabla. Smatrati da se koren nalazi na nivou 0. Na standardni izlaz ispisati rezultat izvršavanja napisane funkcije.

stablo.txt	1 2 3 4 5 6	6 4 3 5 8 7 9	6 4 3 5 8 7 9 2 0	1	-1 -2 -3 -4 -5 -6 -7	<i>prazna datoteka</i>
Izlaz	15	5670	0	1	-105	0