

Ime i prezime

Indeks

Smjer

1	2	3	4	Σ

**PROGRAMIRANJE I**  
**Završni ispit 2022/23.**  
**I grupa**

1. Napisati program koji učitava niz cijelih brojeva **X**, dužine **N**. Program treba da odredi i odštampa prvi element koji je veći od svog prethodnika (prethodni element niza). Ukoliko ne postoji nijedan takav element, program treba da odštampa poruku "Nema ih". (10 poena)

**Primjer:** Za niz  $X=[6,5,3,7,9,1]$ , program treba da odštampa broj 7, a za  $X=[6,5,3,2]$  treba da odštampa poruku "Nema ih".

2. Napisati funkciju koja za argumente ima niz cijelih brojeva **X**, dužine **N**, i pokazivač na string **S**. Funkcija treba da formira string **S** nadovezivanjem elemenata niza **X**, pri čemu se uklanja predznak elemenata i ispred negativnih elemenata dodaje slovo 'n', a ispred pozitivnih slovo 'p'. Nule se ignorišu. Funkcija ne vraća ništa. (9 poena)

Napisati i glavni program koji učitava niz **X**, poziva napisanu funkciju i štampa tako dobijeni string. (3 poena)

**Primjer:** Za niz  $[2,0,-15,13,-5,78]$ , funkcija treba da formira string "p2n15p13n5p78".

3. a) Lista je kreirana u fajlu **Lista.c**. Svaki čvor liste sadrži cio broj. Na poziciji označenoj sa `/**/` pozvati funkciju koja pronalazi najmanji element liste i premješta ga na kraj liste. Ukoliko postoji više takvih elemenata, premjestiti samo prvi. U funkciji `main()` testirati napisanu funkciju. (7 poena)

b) U fajlu **Stablo.c** je kreirano binarno stablo. Svaki čvor stabla sadrži prirodan broj **i**. Napisati funkciju koja vraća **1** ukoliko postoji bar jedna putanja od korijena ka listovima kod koje su svi brojevi parni. U okviru funkcije `main()` testirati napisanu funkciju. (7 poena)

4. U igri Jamb sa 6 kockica, ful predstavlja situaciju u kojoj dobijate 3 iste + 2 iste kockice (npr. 3 petice i 2 dvojke), dok vrijednost šeste kockice nije bitna (bilo koji broj iz  $\{1,2,3,4,5,6\}$ ). Napisati program koji simulira bacanje 6 kockica i procjenjuje i štampa vjerovatnoću dobijanja fula. U tom smislu, generisati **N** bacanja kockica i traženu vjerovatnoću izračunati kao količnik broja dobijenih fulova i broja **N**. (14 poena)

**Napomena:** Bacanje jedne kockice realizovati funkcijom `rand()` (biblioteka `stdlib.h`) koja vraća slučajan cijeli broj iz opsega  $[0, \text{RAND\_MAX}]$  (`RAND\_MAX` je najmanje 32767).

**Napomene:** 1. Da bi se pregledali, zadaci moraju proći fazu kompajliranja.  
2. Prvi zadatak se boduje binarno (sve ili ništa).

Test traje 75 minuta. Nakon završetka ispita, izradu zadataka ćete unijeti u odgovarajuća polja forme na sajtu *distance learning* platforme. Student je obavezan da zna svoje kredencijale za platformu, bez njih polaganje nije moguće.

Ime i prezime

Indeks

Smjer

1	2	3	4	Σ

**PROGRAMIRANJE I**  
**Završni ispit 2022/23.**  
**II grupa**

1. Napisati program koji učitava niz realnih brojeva **X**, dužine **M**. Program treba da odredi i odštampa prvi element koji je veći od svog sljedbenika (naredni element niza). Ukoliko ne postoji nijedan takav element, program treba da odštampa poruku "Nema ih". (10 poena)

**Primjer:** Za niz  $X=[6.1, 7.5, 8.3, 7.2, 9, 1.8]$ , program treba da odštampa broj 8.3, a za  $X=[6.1, 7.5, 7.8]$  treba da odštampa poruku "Nema ih".

2. Napisati funkciju koja za argumente ima niz cijelih brojeva **X**, dužine **N**, i pokazivač na string **S**. Funkcija treba da formira string **S** nadovezivanjem pozitivnih elemenata niza **X**, pri čemu će ispred parnih elemenata dodati slovo 'p', a ispred neparnih slovo 'n'. Funkcija ne vraća ništa. (9 poena)

Napisati i glavni program koji učitava niz **X**, poziva napisanu funkciju i štampa tako dobijeni string. (3 poena)

**Primjer:** Za niz  $[2,27,-15,13,-5,202]$ , funkcija treba da formira string "p2n27n13p202".

3. a) Lista je kreirana u fajlu **Lista.c**. Svaki čvor liste sadrži cio broj. Na poziciji označenoj sa `/**/` pozvati funkciju koja pronalazi najveći element liste i premješta ga na početak liste. Ukoliko postoji više takvih elemenata, premjestiti samo prvi. U funkciji `main()` testirati napisanu funkciju. (7 poena)

b) U fajlu **Stablo.c** je kreirano binarno stablo. Svaki čvor stabla sadrži prirodan broj **i**. Napisati funkciju koja vraća **1** ukoliko postoji bar jedna putanja od korijena ka listovima kod koje su brojevi uređeni u opadajući poredak. U okviru funkcije `main()` testirati napisanu funkciju. (7 poena)

4. U igri Jamb sa 6 kockica, ful predstavlja situaciju u kojoj dobijate 3 iste + 2 iste kockice (npr. 3 petice i 2 dvojke), dok vrijednost šeste kockice nije bitna (bilo koji broj iz  $\{1,2,3,4,5,6\}$ ). Napisati program koji simulira bacanje 6 kockica i procjenjuje i štampa vjerovatnoću dobijanja fula. U tom smislu, generisati **N** bacanja kockica i traženu vjerovatnoću izračunati kao količnik broja dobijenih fulova i broja **N**. (14 poena)

**Napomena:** Bacanje jedne kockice realizovati funkcijom `rand()` (biblioteka `stdlib.h`) koja vraća slučajan cijeli broj iz opsega  $[0, \text{RAND\_MAX}]$  (`RAND\_MAX` je najmanje 32767).

**Napomene:** 1. Da bi se pregledali, zadaci moraju proći fazu kompajliranja.  
2. Prvi zadatak se boduje binarno (sve ili ništa).

Test traje 75 minuta. Nakon završetka ispita, izradu zadataka ćete unijeti u odgovarajuća polja forme na sajtu *distance learning* platforme. Student je obavezan da zna svoje kredencijale za platformu, bez njih polaganje nije moguće.

Ime i prezime

Indeks

Smjer

1	2	3	4	Σ

**PROGRAMIRANJE I**  
**Završni ispit 2022/23.**  
**III grupa**

1. Napisati program koji učitava prirodan broj **K** (ne treba provjeravati). Program treba da odredi i odštampa najmanji i najveći neparan broj sa kojim je djeljiv broj **K**, pri čemu se ne računa broj 1. Ukoliko nema traženih brojeva, odštampati poruku "Nema ih". (10 poena)  
**Primjer:** Za unijeto  $K=30$ , program treba da odštampa brojeve 3 i 15, dok za  $K=4$  treba da prikaže poruku da nema traženih brojeva.
2. Napisati funkciju koja za argument ima string **S** koji predstavlja prirodan broj (ne treba provjeravati). Funkcija kao rezultat treba da vrati broj formiran od neparnih cifara stringa **S**. Ukoliko ne postoji nijedna neparna cifra, funkcija vraća broj 0. (9 poena)  
Napisati i glavni program koji učitava string **S** i testira napisanu funkciju. (3 poena)  
**Primjer:** Za string  $S="984365"$ , funkcija vraća broj 935.
3. a) Lista je kreirana u fajlu **Lista.c**. Svaki čvor liste sadrži cio broj. Na poziciji označenoj sa `/**/` pozvati funkciju koja pronalazi prvi element liste veći od svojih susjeda (prvog ispred i prvog iza) i premješta ga na kraj liste. Krajnje elemente liste ne razmatrati. Ukoliko ne postoji takav element, ne raditi ništa. U funkciji `main()` testirati napisanu funkciju. (7 poena)  
b) U fajlu **Stablo.c** je kreirano binarno stablo. Napisati funkciju koja određuje i vraća koliko ima čvorova čija su oba sina listovi. U okviru glavnog programa testirati napisanu funkciju i odštampati dobijeni rezultat. (7 poena)
4. Učitava se string koji predstavlja sortirani niz prirodnih brojeva, razdvojenih spejsom. Tri ili više uzastopnih brojeva se predstavljaju u obliku intervala "min:max" (npr. brojevi 4 5 6 7 se predstavljaju u obliku "4:7"). Potrebno je formirati niz cijelih brojeva predstavljenih učitanim stringom. Na izlazu štampati dobijeni niz. (14 poena)  
**Primjer:** Ako učitamo string "-5 -3:0 3:5 19 20 22", na izlasku treba štampati niz [-5,-3,-2,-1,0,3,4,5,19,20,22].

**Napomene:** 1. Da bi se pregledali, zadaci moraju proći fazu kompajliranja.  
2. Prvi zadatak se boduje binarno (sve ili ništa).

Test traje 75 minuta. Nakon završetka ispita, izradu zadataka ćete unijeti u odgovarajuća polja forme na sajtu *distance learning* platforme. Student je obavezan da zna svoje kredencijale za platformu, bez njih polaganje nije moguće.

Ime i prezime

Indeks

Smjer

1	2	3	4	Σ

**PROGRAMIRANJE I**  
**Završni ispit 2022/23.**  
**IV grupa**

1. Napisati program koji učitava prirodan broj **M** (ne treba provjeravati). Program treba da odredi i odštampa da li broj **M** ima više parnih ili neparnih djelilaca, ili ih ima isto. (10 poena)  
**Primjer:** Za unijeti broj  $M=40$ , program treba da odštampa obavještenje "Više parnih djelilaca" jer je broj 40 djeljiv sa 1, 2, 4, 5, 8, 10, 20 i 40.
2. Napisati funkciju koja za argument ima string **S** koji predstavlja prirodan broj (ne treba provjeravati). Funkcija kao rezultat treba da vrati broj formiran od parnih cifara stringa **S**. Ukoliko ne postoji nijedna parna cifra, funkcija vraća broj -1. (9 poena)  
Napisati i glavni program koji učitava string **S** i testira napisanu funkciju. (3 poena)  
**Primjer:** Za string  $S="984365"$ , funkcija vraća broj 846.
3. a) Lista je kreirana u fajlu **Lista.c**. Svaki čvor liste sadrži cio broj. Na poziciji označenoj sa `/**/` pozvati funkciju koja pronalazi prvi element liste manji od svojih susjeda (prvog ispred i prvog iza) i premješta ga na početak liste. Krajnje elemente liste ne razmatrati. Ukoliko ne postoji takav element, ne raditi ništa. U funkciji `main()` testirati napisanu funkciju. (7 poena)  
b) U fajlu **Stablo.c** je kreirano binarno stablo. Napisati funkciju koja određuje i vraća koliko ima čvorova koji imaju jednog sina koji je list. U okviru glavnog programa testirati napisanu funkciju i odštampati dobijeni rezultat. (7 poena)
4. Učitava se string koji predstavlja sortirani niz prirodnih brojeva, razdvojenih spejsom. Tri ili više uzastopnih brojeva se predstavljaju u obliku intervala "min:max" (npr. brojevi 4 5 6 7 se predstavljaju u obliku "4:7"). Potrebno je formirati niz cijelih brojeva predstavljenih učitanim stringom. Na izlazu štampati dobijeni niz. (14 poena)  
**Primjer:** Ako učitamo string "-5 -3:0 3:5 19 20 22", na izlasku treba štampati niz [-5,-3,-2,-1,0,3,4,5,19,20,22].

**Napomene:** 1. Da bi se pregledali, zadaci moraju proći fazu kompajliranja.  
2. Prvi zadatak se boduje binarno (sve ili ništa).

Test traje 75 minuta. Nakon završetka ispita, izradu zadataka ćete unijeti u odgovarajuća polja forme na sajtu *distance learning* platforme. Student je obavezan da zna svoje kredencijale za platformu, bez njih polaganje nije moguće.