

Strukture podataka i algoritmi 1

Kolokvijum (40 poena)

Vreme izrade: 150 minuta

Na Desktop-u u direktorijumu **Rad** kreirati direktorijum **ImePrezime_Index** i unutar njega sačuvati fajlove koji sadrže kodove i rešenja datih zadataka. Rešenje prvog zadatka sačuvati u fajlu sa nazivom **zadatak1.c**, rešenje drugog zadatka u fajlu **zadatak2.c**, a rešenje trećeg zadatka u fajlu **zadatak3.c**. Zbog testiranja rešenja zadataka, **obavezno je da izlazi budu u istom formatu kao što su zadati u postavci**.

Zadatak 1

Mika radi kao stručnjak za sajber bezbednost i njegov zadatak je da detektuje hakerske napade na mrežu svoje kompanije. Napadi se ne mogu lako predvideti, ali ih je moguće otkriti analizom logova sistema.

Logovi sistema predstavljaju tekst koji se sastoji od:

- malih i velikih slova engleske abecede
- cifara (0-9)
- specijalnih karaktera (!, @, #, \$, %)

Mika učitava podatke o jednom logu karakter po karakter sve do oznake za novi red. Uočio je da specijalni karakteri predstavljaju ograničavače grupa — svaka grupa počinje i završava se specijalnim karakterom, a sadržaj između njih čine slova i cifre. Karakteri van grupa (između zatvarajućeg i otvarajućeg specijalnog karaktera naredne grupe) se tretiraju kao šum i ignorišu se.

Kako bi pomogli Miki u njegovoj detekciji hakerskih napada, napisati C program koji:

- Učitava tekst karakter po karakter do oznake za novi red i deli ga u grupe na osnovu specijalnih karaktera. Maksimalan broj grupa koji se može pojaviti je 100, a karaktera u jednoj grupi 255.
- Iz svake grupe izdvaja slova (velika slova konvertuje u mala) i dinamički alokira string koji ih čuva (moguće je odraditi dinamičku alokaciju prostora za maksimalni broj karaktera u grupi).
- Za svaku grupu računa zbir cifara koje se u njoj pojavljuju
- Sve izdvojene reči spaja u jedan dinamički alokirani string, redom kojim se pojavljuju u logu. Takođe, moguće je odraditi dinamičku alokaciju prostora za maksimalan broj karaktera koji se može naći u spojenom stringu.
- Rekurzivnom funkcijom proverava da li je spojeni string palindrom
- Na kraju oslobađa svu dinamički alociranu memoriju

Detekcija napada funkcioniše po sledećem pravilu: *Ukoliko je spojeni string palindrom i ukupan zbir svih cifara iz loga je deljiv sa 5, ispisati upozorenje o mogućem hakerskom napadu. U suprotnom, log je bezbedan.*

Napomene:

- Svaka grupa je oblika <specijalni karakter><sadržaj><specijalni karakter>
- Karakter koji se nalaze van grupa se ignorišu
- Potpis rekurzivne funkcije koju je potrebno implementirati mora izgledati ovako: **int jePalindrom(char*, int, int)**.
- Po završetku, potrebno je osloboditi svu alociranu memoriju.

Primer

Ulaz: #m2a!%3na##1#!a35

Objašnjenje parsiranja:

Sadržaj između ograničavača	Izdvojena slova	Zbir cifara u grupi
#m2a!	ma	2
%3na#	na	3
#1#	(prazna)	1
!a35 – van grupe, ignoriše se		

Izlaz:

```

Grupa 1: slova = "ma", zbir cifara = 2
Grupa 2: slova = "na", zbir cifara = 3
Grupa 3: slova = "", zbir cifara = 1

Spojena rec: "mana"
Ukupan zbir cifara: 6

Spojena rec NIJE palindrom.
Ukupan zbir cifara NIJE deljiv sa 5.
=> Sistem je bezbedan.
```


4 2 700
Jelena Markovic
3 4 200
Milan Stojanovic
2 3 500
Lenka Milic
4 5 50
Petar Nesic
4 4 400

Izlaz:

Za gosta Lenka Milic nije nadjen odgovarajuci hotel.
Vila Mirta, broj osoba: 4
Jadran, broj osoba: 2
Sunrise Beach Hotel, broj osoba: 9
Palma Resort, broj osoba: 1
Hotel sa najvećim prihodom od 769.50 je: Sunrise Beach Hotel