

UNIVERZITET U TUZLI  
FAKULTET ELEKTROTEHNIKE



# Strukture podataka

Liste

Zadaća 1

Tuzla, april/travanj 2019.

Franjevačka 2, 75000 Tuzla, Bosna i Hercegovina  
Web: <http://fet.ba/>

# Sadržaj

<b>Sadržaj</b>	<b>2</b>
<b>Zadatak 1:</b>	<b>3</b>
<b>Zadatak 2:</b>	<b>4</b>
2.1. Dohvatanje informacija za cijelu godinu	4
2.2. Dohvatanje informacija za određeni mjesec	5
2.3. Dodavanje informacija	5
2.4. Napomene	5

## Zadatak 1:

- Napraviti sortiranu dvostruko linkanu listu. Lista treba da ima default, copy i move konstruktore, te copy i move operatore jednako. Lista treba da sadrži metode add(), find() i remove(). Metodi find() i remove() rade sa iteratorima, tako da je potrebno implementirati i iteratore za listu. Metod add() dodaje element na poziciju tako da lista bude sortirana, metod find() uzima predikat i vraća iterator na element koji zadovoljava predikat funkciju, a ukoliko ni jedan element ne zadovoljava, vraća iterator na kraj (end()). Metod remove() uzima iterator i briše element u listi sa te pozicije. (Obratiti pažnju na granične slučajeve!).  
Napomena: potrebno je da lista bude generička.
- Napraviti strukturu ToDoTask, koja predstavlja obavezu koju korisnik treba uraditi. ToDoTask se sastoji od id-a, naslova (title), opisa (description) te prioriteta (priority). Implementirati potrebne operatore za dodavanje strukture u prethodno implementiranu sortiranu listu. Prilikom kreiranja objekta ToDoTask, od korisnika traziti unos naslova, opisa i prioriteta, a za vrijednost id-a iskoristiti sljedeću generator funkciju:

```
int generateId(){  
    static int counter = 0;  
    return ++counter;  
}
```

- Napraviti glavni program koji implementira funkcionalnost notesa. Korisniku je potrebno prikazati meni za unos nove obaveze, prikaz trenutnih obaveza te označavanje obaveze kao završene. Obaveze se onose u sortiranu listu, tako da se obaveze sa najvećim prioritetom nalaze na početku liste. Prilikom označavanja obaveze kao završene, korisniku ponuditi unos id-a obaveze, te na osnovu toga izvršiti brisanje obaveze iz liste (koristeći metode find() i remove()).

## Zadatak 2:

Napisati program koji korisniku daje historijske meteorološke informacije za određenu meteorološku stanicu. File `stations.csv` sadrži listu meteoroloških stanica za koje su podaci dostupni. Za svaku meteorološku stanicu dat je njen jedinstven identifikator, geografska širina, geografska dužina, nadmorska visina, grad/mjesto u kojem se stanica nalazi te dodatan csv file u kojem se nalaze svi zabilježeni podaci o minimalnoj i maksimalnoj temperaturi koju je ta meteorološka stanica zabilježila za određeni mjesec.

Kada se program pokrene treba korisniku da izlista meni kao na slici 2.1.

```
***** Historical temperature information *****
1. Get information for entire year
2. Get information for specific month
3. Insert new historical temperature information
4. Exit
Your choice?: █
```

Slika 2.1. Inicijalni meni

### 2.1. Dohvatanje informacija za cijelu godinu

Ukoliko je odabrana opcija 1, korisnik dalje treba da unese lokaciju stanice koja ga zanima nakon čega program ispisuje informacije o toj stanici kao što je to prikazano na slici 2.2.

```
Enter location where station is placed: Tuzla
StationID: BKM00014557
Latitude: 44.55
Longitude: 18.7
Elevation: 307
Enter year: █
```

Slika 2.2. Izbor 1

Nakon unošenja godine program treba da ispiše sve mjesece te godine zajedno sa minimalnom i maksimalnom temperaturom tog mjeseca. Ukoliko unutar file-a, taj mjesec nije dostupan program treba da ispiše N/A (not available) za minimalnu i maksimalnu temperaturu (Slika 2.3.).

```
Enter year: 2002

Month  Max temp  Min temp
Jan    5.4      -4.83
Feb    13.78    0.38
Mar    15.66    2.73
Apr    17.01    5.22
May    N/A     10.9
Jun    N/A     N/A
Jul    N/A     15.36
Aug    N/A     N/A
Sep    N/A     N/A
Oct    19.08    6.34
Nov    15.86    N/A
Dec    N/A     N/A
```

Slika 2.3. Konačan ispis prvog dijela programa

## 2.2. Dohvatanje informacija za određeni mjesec

Opcija 2 treba također da korisniku ponudi unošenje lokacije, te da ispiše informacije o toj stanici (Slika 2.2.). Nakon toga korisnik treba da unese godinu te mjesec za koji želi dohvatiti informacije te program ispisuje minimalnu i maksimalnu temperaturu za traženi mjesec (Slika 2.4.).

```
Enter location where station is placed: Tuzla
StationID: BKM00014557
Latitude: 44.55
Longitude: 18.7
Elevation: 307

Enter year: 2008
Enter month: 5
Max temp      Min temp
24.24         8.94
```

Slika 2.4. Konačan ispis drugog dijela programa

Kao i u prvom dijelu, ukoliko informacije nisu dostupne program treba da ispiše N/A.

## 2.3. Dodavanje informacija

Potrebno je implementirati i opciju kojom dodajemo novu informaciju o temperaturi za određeni mjesec. Od korisnika se traži unos jedinstvenog identifikatora stanice, godina, mjesec te minimalna i maksimalna temperatura. Ukoliko stanica sa unešenim identifikatorom ne postoji, program treba da ispiše da stanica ne postoji i vrati se na početni meni. Nakon unošenja podataka, korištenjem opcija 1 i 2 testirati da li smo pravilno unijeli podatke u korištene strukture podataka. U slučaju da informacija za unešeni mjesec već postoji, potrebno je prebrisati staru vrijednost. Nije potrebno ove informacije ponovno snimati u odgovarajuće fajlove već dodane informacije postoje samo dok program živi.

## 2.4. Napomene

Za učitavanje fajlova koristiti `std::ifstream` objekat iz standardne biblioteke. Za organizaciju podataka nije potrebno implementirati nikakve kolekcije već odabrati najefikasnije strukture podataka za pretraživanje (prvi i drugi dio programa) iz standardne biblioteke (od onih koje smo do sada radili). CSV file-ovi se nalaze u prilogu zadatac (zadatak2.zip).