

Fakultet tehničkih nauka, DRA, Novi Sad

Predmet:  
Organizacija podataka

Dr Slavica Kordić  
Dr Vladimir Ivančević  
Vladimir Ivković  
Nikola Todorović

# Biblioteka Struct u programskom jeziku *Python*

# **STRUKTURIRANE BINARNE DATOTEKE**

# *struct* tip u programskom jeziku C

- Strukture (slogovi) u programskom jeziku C predstavljaju složeni, korisnički definisan tip podataka
- Komponente (polja) strukture su imenovana i mogu biti različitog tipa
- Primer definisanja strukture:

```
struct student {  
    char index[12];  
    char ime[16];  
    char prezime[16];  
    int godinaStudija;  
};
```

# Modul *struct*

- Modul *struct* omogućava konverziju između Pajton objekata i C struktura koje su predstavljene kao *bytes* objekti
- Koriste se formatni stringovi za opis načina pakovanja/raspakivanja vrednosti promenljivih (polja strukture) u bajtove
- Koristeći *struct* modul ,moguće je binarnu datoteku kreirati u Pajtonu, a čitati u C-u i obrnuto

# Modul *struct*

- Glavne funkcije *struct* modula su:
  - *struct.pack(format, v1, v2, ...)* koristi se za pakovanje vrednosti promeljivih *v1, v2, ...* na način specificiran formatnim stringom u niz bajtova
  - *struct.unpack(format, buffer)* vraća torku (*tuple*) dobijenu raspakivanjem promenljive *buffer*
  - *struct.calcsize(format)* određuje veličinu strukture (u bajtovima) opisane formatnim stringom

# Formatni string

- Formatnim stringom se specificira na koji način će vrednosti promenljivih biti spakovane/raspakovane u/iz niza bajtova
- Primer formatnog string:
  - `pack('hh1', 1, 2, 3)`
  - `pack('ci', b'*', 0x12131415)`
  - `pack('i7s', 1234, b"test")`
  - `calcsiz('ci')`
- Kada formatni string specificira čuvanje string vrednosti, mora se eksplicitno naglasiti maksimalni broj karaktera u stringu
  - Zbog načina na koji se stringovi čuvaju u C-u, ovu vrednost bi uvek trebalo dodatno **uvećati za 1**
    - Prostor za karakter `\0`

# Formatni string

Format	C Type	Python type	Standard size
x	pad byte	no value	
c	char	bytes of length 1	1
b	signed char	integer	1
?	_Bool	bool	1
h	short	integer	2
i	int	integer	4
l	long	integer	4
q	long long	integer	8
f	float	float	4
d	double	float	8
s	char[]	bytes	
p	char[]	bytes	



# Primer 1

- Zadata CSV datoteku pročitati i njen sadržaj smestiti u binarnu datoteku
  - slogovi bi trebalo da odgovaraju redovima ulazne datoteke
- Kolone iz CSV datoteke:
  - Username
  - Identifier
  - First name
  - Last name

# Primer 2

- Binarnu datoteku, formiranu u prethodnom primeru, pročitati i ispisati sadržaj 2. i 5. sloga

# Zadatak 1

- Niz 2D tačaka dat je u ulaznoj tekstualnoj datoteci:

1.23 -6.87

12.45 -0.56

34.55 82.04

- Učitati dati niz i njegov sadržaj sačuvati u izlaznu binarnu datoteku, s tim što bi na početku datoteke trebalo da se nađe i zaglavlje sa dve celobrojne vrednosti:
  - broj tačaka,
  - dimenzionalnost tačaka (u ovom slučaju 2)

# Zadatak 2

- Binarnu datoteku kreiranu u prethodnom zadatku učitati koristeći podatke iz zaglavlja tj. bez prethodnog znanja o broj tačaka i njihovoj dimenzionalnosti
- Izračunati težište (centroid) niza tačaka
- Izmeniti Zadatak 1 tako da radi sa nizom 3D tačaka, pa testirati rešenje Zadatka 2 sa novom binarnom datotekom

# Zadatak 3

- Pročitati sadržaj binarne datoteke *username.bin* na sledeći način:
  - Smatrati da je datoteka podeljena u blokove, s faktorom blokiranja 3
  - Pročitati drugi blok, (koji čine poslednja 3 sloga)
  - Iz pročitano bloka izdvojiti podatke o slogovima, dekodirati ih, i prikazati na ekranu
  - Očekivani ispis programa:

```
['jenkins46', 9346, 'Mary', 'Jenkins']  
['smith79', 5079, 'Jamie', 'Smith']  
['vlad58', 5858, 'Vlad', 'Nevski']
```