

# Основи програмирања

## Вежбе 2

Исидора Грујић  
isidora@uni.kg.ac.rs

Лазар Илић  
iliclazar15@gmail.com

Факултет инжењерских наука  
Универзитет у Крагујевцу

октобар 2023.

- 1 Кратак преглед
- 2 Нескаларни објекти: ниске
- 3 Гранање
- 4 Понављање
- 5 Задаци

- 1 Кратак преглед
- 2 Нескаларни објекти: ниске
- 3 Гранање
- 4 Понављање
- 5 Задаци

- врсте програмских језика
- Пајтон 3/Python 3, инсталација, окружење
- учитавање података са стандардног улаза/исписивање на стандардном излазу
- типови објеката - скаларни објекти, претварање типова
- аритметички оператори над бројевима, оператори поређења над бројевима, уграђене математичке функције
- логички оператори
- дефинисање функција

- 1 Кратак преглед
- 2 Нескаларни објекти: ниске
- 3 Гранање
- 4 Понављање
- 5 Задаци

# Нескаларни објекти: ниске

- сложени објекти
- имају унутрушњу структуру којој се може приступати
- **ниске/str** - најједноставнији нескаларни објекти
- низови алфанумеричких знакова

## Пример 1

Разни примери ниски

## Пример 2

Учитавање података са улаза - тип објекта

# Исписивање на стандардном излазу

## Пример 3

Функција **print()** - навођење променљивих у витичастим заградама

## Пример 4

Функција **print()** - одвајање делова реченице и променљивих запетама

## Пример 5

Функција **format()** vs функција **round()**

# Оператори над нискама

- "аритметички" оператори
  - сабирање: `<прва_ниска> + <друга_ниска>`
  - множење целим бројем: `<цео_број> * <ниска>`

## Пример 6

Функција **print()** - "додавање" променљивих у реченицу

- уграђене функције
  - функција **len()**
  - функција **lower()**
  - функција **upper()**
  - функција **islower()**
  - функција **isupper()**
  - функција **strip()**
  - функција **lstrip()**
  - функција **rstrip()**
- оператори поређења



# Извлачење делова ниски

- индексирање:  $\langle \text{ниска} \rangle [\langle \text{индекс} \rangle]$

## Пример 7

Извлачење делова ниски индексирањем - потенцијална грешка.

- одсецање:
  - $\langle \text{ниска} \rangle [\langle \text{почетак} \rangle : \langle \text{крај} \rangle]$
  - $\langle \text{ниска} \rangle [\langle \text{почетак} \rangle : \langle \text{крај} \rangle : \langle \text{корак} \rangle]$

## Пример 8

Извлачење делова ниски одсецањем - "обртање" ниске

- 1 Кратак преглед
- 2 Нескаларни објекти: ниске
- 3 Гранање**
- 4 Понављање
- 5 Задачи

- програм се не извршава увек праволинијски
- најједноставнија наредба гранања је прост услов
- провера услова који као резултат даје објекат типа `bool`
  - услов је логички израз
- у логичком језику Пајтон гранање је имплементирано помоћу наредбе `if`

- дефинисање блока наредби - **if** <логички\_израз>:  
наредбе које желимо да се извршавају у случају да је испуњен услов представљен логичким изразом
- опционо - додатни услови - **elif** <други\_логички\_израз>:  
наредбе које желимо да се извршавају у случају да је испуњен услов представљен логичким изразом у другом условном блоку
- коначно - **else**:  
наредбе које желимо да се извршавају у случају да ниједан од услова постављених логичким изразима није испуњен

Гранање може бити угнеждено (енгл. *nested*).

- 1 Кратак преглед
- 2 Нескаларни објекти: ниске
- 3 Гранање
- 4 Понављање**
- 5 Задаци

- неопходан је један или више начина за писање програма произвољне сложености
- имплементација понављања користећи **while** петљу
- дефинисање тела петље - **while** <логички\_израз>:  
наредбе које желимо да се извршавају у итерацијама, све док је задовољен услов представљен логичким изразом

- 1 Кратак преглед
- 2 Нескаларни објекти: ниске
- 3 Гранање
- 4 Понављање
- 5 Задаци**

Задатке са првих вежби можете пронаћи на [страници предмета](#).