

# Grananja, liste

Softverske i informacione tehnologije, 1. godina OSS

# Relacione operacije i operatori

- Relacione operacije služe za poređenje vrednosti. Rezultat poređenja vrednosti je logičkog tipa (True, False).

Znak operatora	Objašnjenje	primer
<	Utvrdjuje da li je prvi operand manji od drugog	$4 < 11 \Leftrightarrow \text{True}$
>	Utvrdjuje da li je prvi operand veci od drugog	$3.4 > 2 \Leftrightarrow \text{False}$
<=	Utvrdjuje da li je prvi operand manji ili jednak drugom	$3.4 \leq 2 \Leftrightarrow \text{False}, 1 \leq 1 \Leftrightarrow \text{True}$
>=	Utvrdjuje da li je prvi operand veci ili jednak drugom	$10 \geq 11 \Leftrightarrow \text{False}, 2 \geq 2 \Leftrightarrow \text{True}$
==	Proverava jednakost operanada	$2 == 3 \Leftrightarrow \text{False}, 3 == 3 \Leftrightarrow \text{True}$
!=	Proverava da li su operandi razliciti	$2 != 3 \Leftrightarrow \text{True}, 3 != 3 \Leftrightarrow \text{False}$

# Logičke operacije i operatori

- Logičke operacije pomažu pri donošenju odluka na osnovu postojećih uslova. I operandi i rezultat logičkih operacija su logičkog tipa.

Znak operatora	Objašnjenje	primer
and	Rezultat je tačan ako su oba operanda tačna.	$3 \neq 2$ and $1 < 3 \Leftrightarrow \text{True}$
or	Rezultat je tačan ako je bilo koji od operanada tačan.	$3 == 2$ or $1 < 3 \Leftrightarrow \text{True}$
not	Menja logičku vrednost na suprotnu.	$\text{not}(5 \neq 5.1) \Leftrightarrow \text{False}$

# Tablice istinitosti

x	y	x AND y
T	T	T
T	F	F
F	T	F
F	F	F

x	y	x OR y
T	T	T
T	F	T
F	T	T
F	F	F

x	NOT x
T	F
F	T

# Grananja

Grananja se koriste ukoliko postoji potreba da se delovi koda ne izvršavaju uvek, već samo ukoliko je neki uslov ispunjen.

## Osnovni oblik

```
if expression:  
    statement  
  
else:  
    statement
```

## Prošireni oblik

```
if expression:  
    statement  
elif expression:  
    statement  
...  
else:  
    statement
```

# Liste

- Lista je sekvenca podataka
- Liste ne moraju da budu homogene! Lista može da sadrži različite tipove podataka: brojeve, stringove, objekte...
- Liste su uređene i imaju konačan broj elemenata
- Elementi liste su indeksirani u odnosu na udaljenost od prvog elementa
- Prvi element liste je na indeksu 0!

# Iteriranje kroz listu

```
list = ['physics', 'chemistry',  
1997, 2000]
```

```
for item in list:  
    print(item)
```

```
list = ['physics', 'chemistry', 1997,  
2000]
```

```
number_of_elements = list.count()
```

```
for i in range(number_of_elements):  
    print(list[i])
```



# Operacije nad listama

```
list = ['apple', 'banana', 'cherry']
```

<code>list.pop(1)</code>	Uklanja element sa određene pozicije
<code>list.reverse()</code>	Obrće redosled elemenata
<code>list.sort()</code>	Sortira elemente liste rastuće po alfabetu
<code>list.insert(2, 'orange')</code>	Ubacuje novi element u listu, na zadatom indeksu
<code>list.append('orange')</code>	Dodaje element na kraj liste
<code>new_list = list.copy()</code>	Pravi kopiju liste
<code>list.clear()</code>	Uklanja sve elemente iz liste