

Napredno programiranje i programski jezici

10 Python

Fakultet tehničkih nauka, Novi Sad
23-24/Z
Dunja Vrbaški

```
a = 5  
b = 0  
c = 3
```

```
print(a and b)  
print(a or b)
```

```
print(a and c)  
print(a or c)
```

```
0  
5  
3  
5
```

Svaki objekat može učestvovati u logičkim operacijama ili kao uslov u if i while iskazima

```
a = 5  
b = 3  
print(a, b)
```

```
5 3
```

```
a = 5  
b = 3  
c = "abc"  
print(a, b, c)
```

```
5 3 abc
```

```
a = 5  
b = 3  
c = "abc"  
print(a, b, c, "dobar dan")
```

```
5 3 abc dobar dan
```

```
a = 5  
b = 3  
c = "abc"  
print(a, b, c)
```

5 3 abc

```
a = 5  
b = 3  
c = "abc"  
print(a, b, c, sep = ", ")
```

5, 3, abc

```
a = 5  
b = 3  
c = "abc"  
print(a, b, c, sep = ", ", end = ".")
```

5, 3, abc.

```
a = 5
b = 3
c = "abc"
print(a + b + c)
```

TypeError: unsupported operand
type(s) for +: 'int' and 'str'

```
a = 5
b = 3
c = "abc"
print(c + "dobar dan")
```

abcdobardan

```
a = 5
b = 3
c = "abc"
print(c + str(a))
```

abc5

```
a = 5
b = 3
print ("zbir je", a + b)
```

zbir je 8

```
a = 5
b = 3
print("zbir vrednosti", a, "i", b, "je", a + b)
```

zbir vrednosti 5 i 3 je 8

```
a = 5
b = 3
print(f"zbir vrednosti {a} i {b} je {a + b}")
```

zbir vrednosti 5 i 3 je 8

postoji i C- printf-style formatiranje (%)
postoji i metod format()

```
ime = input("Unesite ime...")  
print("Pozdrav", ime)
```

```
Unesite ime...
```

```
Unesite ime...Harry
```

```
Unesite ime...Harry  
Pozdrav Harry
```

```
ime = input("Unesite ime...")  
godine = input("Unesite godine...")  
godiste = 2023-godine  
print(f"{ime}, godiste: {godiste}")
```

Unesite ime...

?


```
ime = input("Unesite ime...")
godine = input("Unesite godine...")
godiste = 2023-godine
print(f"{ime}, godiste: {godiste}")
```

```
Unesite ime...Petar
Unesite godine...20
Traceback (most recent call last):
  File "d:\NPPJPy\print.py", line 3, in <module>
    rodjen = 2023-godine
TypeError: unsupported operand type(s) for -:
'int' and 'str'
```

```
ime = input("Unesite ime...")
godine = input("Unesite godine...")
godiste = 2023-int(godine)
print(f"{ime}, godiste: {godiste}")
```

```
Unesite ime...Petar
Unesite godine...20
Petar, godiste: 2003
```

```
a = "dobar dan"
```

```
print(a + a)
```

```
print(a, a)
```

```
dobar dandobar dan  
dobar dan dobar dan
```

```
a = "dobar dan"
```

```
print(a * 3)
```

```
print(a, a, a, sep = )
```

```
dobar dandobar dandobar dan  
dobar dandobar dandobar dan
```

```
a = "dobar dan"  
print(len(a))  
print(a[3])  
print(a[3:5])
```

```
9  
a  
ar
```

Obrati pažnju: poslednji nije uključen

```
a = "dobar dan"

print(a[3])

print(a[3:5])

print(a[3:len(a)])
print(a[3:])

print(a[0:5])
print(a[:5])
```

```
a
ar
ar dan
ar dan
dobar
dobar
```

```
a = "dobar dan"
```

```
print(a[3])
```

```
a[3] = "x"
```

```
TypeError: 'str' object does not support item  
assignment
```

```
a = "dobar dan"  
  
print(a.upper())  
print(a)  
print(a.isupper())
```

```
DOBAR DAN  
dobar dan  
False
```

```
print(a.upper())  
print(a)  
print(a.isupper())
```

a.

- find
- format
- format_map
- index
- # isalnum
- # isalpha
- isascii
- # isdecimal
- # isdigit
- isidentifier
- # islower
- # isnumeric




```
a = "dobar dan"  
print("dan" in a)  
print("dna" not in a)  
print("dan" not in a)
```

```
True  
True  
False
```

```
a = 5
```

```
b = 3
```

```
if (a > b):
```

```
    print(f"{a} je veće od {b}")
```

```
5 je veće od 3
```

```
a = 5  
b = 3
```

```
if (a > b):  
    print(f"{a} je veće od {b}")
```

5 je veće od 3

```
a = 5  
b = 3
```

```
if (a > b):  
print(f"{a} veće od {b}")
```

```
if.py > ...  
1 a = 5  
2 b = 3  
3 if (a > b):  
4     print(f"{a} veće od {b}")  
5
```

IndentationError: expected an indented block
after 'if' statement on line 3

```
a = 3
```

```
b = 5
```

```
if (a > b):
```

```
    print(f"{a} je veće od {b}")
```

```
else:
```

```
    print(f"{a} je manje ili jednako od {b}")
```

3 je manje ili jednako od 5

Suprotno (negacija) od $>$ je \leq

```
a = 3
```

```
b = 5
```

```
if (a > b):
```

```
    print(f"{a} je veće od {b}")
```

```
elif (a == b):
```

```
    print(f"{a} je jednako sa {b}")
```

```
else:
```

```
    print(f"{a} je manje od {b}")
```

3 je manje od 5

C++: else if ?

```
a = 4

if (a > 10):
    s = f"{a} je veće od 10"
elif (a > 5):
    s = f"{a} je: 6, 7, 8, 9 ili 10"
elif (a > 0):
    s = f"{a} je: 1, 2, 3, 4 ili 5"
elif (a == 0):
    s = f"{a} je: 0"
else:
    s = f"{a} je negativno"

print(s)
```

4 je: 1, 2, 3, 4 ili 5

*if - može biti ugnježđeno
Postoji skraćeni oblik*

```
a = 4

if ((a % 3 == 0) and (a % 2 == 0)):
    print(f"broj {a} je deljiv i sa dva i sa 3")
else:
    print(f"broj {a} nije zajednički sadržalac brojeva 2 i 3, \n
ali je možda sadržalac nekog od njih")
```

```
if ((a % 3 == 0) and (a % 2 == 0)):
    print(f"broj {a} je deljiv i sa dva i sa 3")
else:
    s = f"broj {a} nije zajednički sadržalac brojeva 2 i 3, " \
        " ali je možda sadržalac nekog od njih"
    print(s)
```

broj 4 nije zajednički sadržalac brojeva 2 i 3, ali je možda sadržalac nekog od njih

```
a = 4
```

```
if ((a % 3 == 0) and (a % 2 == 0)):  
    print(f"broj {a} je deljiv sa 6")  
else:  
    print(f"broj {a} nije deljiv sa 6")
```

broj 4 nije deljiv sa 6

~~&&
||
!~~

and
or
not

logički

!=

relacioni

&
|

nad bitovima


```
a = 4
```

```
if (0 < a < 5):  
    print(f"broj {a} je veći od 0 i manji od 5")
```

```
broj 4 je veći od 0 i manji od 5
```

$(0 < a) \text{ and } (a < 5)$

```
a = 0
while a < 5:
    print(a)
    a += 1
```

```
0
1
2
3
4
```

```
a, b = 0, 1
while a < 10:
    print(a)
    a, b = b, a + b
```

```
0
1
1
2
3
5
8
```

```
msg = "Unesite tekst ili 'kraj' za prekid:"
s = input(msg)

while (s != "kraj"):
    print(s[::-1])
    s = input(msg)
```

?

```
msg = "Unesite tekst ili 'kraj' za prekid:"
print(msg[1])
print(msg[-1])

print(msg[:])
print(msg[:2])
print(msg[::-1])
print(msg[::-2])
```

```
n
:
Unesite tekst ili 'kraj' za prekid:
Ueietktii'rj apei:
:dikerp az 'jark' ili tsket etisenU
:iepa jr'iitkteieU
```

*string - može obe vrste navodnika
\\ - ako koristimo isti*

```
for i in range(10):  
    print(i)
```

```
for i in range(0, 10):  
    print(i)
```

```
0  
1  
2  
3  
4  
5  
6  
7  
8  
9
```

Obrati pažnju: poslednji nije uključen

[...)

```
for i in range(2, 10, 2):  
    print(i)
```

```
2  
4  
6  
8
```

```
for i in range(10, 2, -2):  
    print(i)
```

```
10  
8  
6  
4
```

```
s = "dobar dan"
```

```
for i in range(0, len(s)):  
    print(s[i])
```

```
s = "dobar dan"
```

```
for ch in s:  
    print(ch)
```

```
d  
o  
b  
a  
r  
  
d  
a  
n
```

ch - string duzine 1

- petlje mogu imati `break` i `continue`
- petlje mogu imati `else`
- postoji `pass`
- postoji `match-case` (kao switch)


```
def zbir(a, b) :  
    return a + b  
  
print(zbir(3, 5))
```

8

```
def paran(a) :  
    return a % 2 == 0  
  
print(paran(5))
```

False

```
import math
```

```
def povrsinaKrug(r) :  
    return r ** 2 * math.pi
```

```
print(povrsinaKrug(3))
```

```
28.274333882308138
```

```
import math
```

```
def povrsinaKrug(r) :  
    return r ** 2 * math.pi
```

```
print("{:.2f}".format(povrsinaKrug(3)))
```

```
28.27
```

```
from datetime import date
```

```
def danas() :  
    return date.today()
```

```
print(danas())
```

2023-12-10

```
def printZbir(a, b) :  
    print(a + b)  
  
printZbir(3, 5)
```

8

```
def printZbir(a, b) :  
    print(a + b)  
  
a = printZbir(3, 5)  
print(a)
```

8

None

```
def zbir(a, b) :  
    return a + b  
  
def paran(a) :  
    return a % 2 == 0  
  
print(paran(zbir(3, 5)))
```

True

```
def zbir(a, b) :  
    return a + b  
  
print(zbir(3, 5))  
print(zbir(b = 5, a = 3))
```

```
8  
8
```

args vs kwargs (positional vs keyword)

```
def sum(maxNum):  
    s = 0  
    for n in range(maxNum + 1):  
        s += n  
    return s  
  
print(sum(5))
```

15

```
def sum(*nums):  
    s = 0  
    for n in nums:  
        s += n  
    return s  
  
print(sum(1, 2, 3, 4, 5))  
print(sum(11, -2, 3, 4, 500))
```

15
516

*args


```
def min(*nums):  
    m = nums[0]  
    for n in nums:  
        if (n < m):  
            m = n  
    return m  
  
print(min(2, 3, 1, 5))
```

1

```
def useParams(**p) :  
    if "mean" in p:  
        print(f"Srednja vrednost je {p['mean']}")  
  
useParams(mean = 15, std = 1)
```

15

****kwargs**

ZADATAK

Isprobati sve primere.

Implementirati poznato (staro) pitanje sa razgovora za posao: FizzBuzz.

Vratiti se na args/kwargs nakon lista i rečnika.