

INFORMATIČKI KLUB

FUTURA

LIGA PROGRAMIRANJA



python

#3

LIGA PROGRAMIRANJA U PYTHONU ZA

OSNOVNE ŠKOLE – 2. RADIONICA

Tomo Sjekavica, Mario Miličević *Informatički klub FUTURA*
Dubrovnik, 05. studenog 2016.



Dubrovnik

Zaštićeno licencom <http://creativecommons.org/licenses/by-nc-sa/3.0/hr/>

Creative Commons



- **slobodno smijete:**

- **dijeliti** — umnožavati, distribuirati i javnosti priopćavati djelo
- **remiksirati** — prerađivati djelo



- **pod slijedećim uvjetima:**

- **imenovanje.** Morate priznati i označiti autorstvo djela na način kako je specificirao autor ili davatelj licence (ali ne način koji bi sugerirao da Vi ili Vaše korištenje njegova djela imate njegovu izravnu podršku).
- **nekomercijalno.** Ovo djelo ne smijete koristiti u komercijalne svrhe.
- **dijeli pod istim uvjetima.** Ako ovo djelo izmijenite, preoblikujete ili stvarate koristeći ga, preradu možete distribuirati samo pod licencom koja je ista ili slična ovoj.



U slučaju daljnjeg korištenja ili distribuiranja morate drugima jasno dati do znanja licencne uvjete ovog djela. Najbolji način da to učinite je linkom na ovu internetsku stranicu.

Od svakog od gornjih uvjeta moguće je odstupiti, ako dobijete dopuštenje nositelja autorskog prava.

Ništa u ovoj licenci ne narušava ili ograničava autorova moralna prava.

Tekst licence preuzet je s <http://creativecommons.org/>.

Raspored Lige programiranja

- 22.10.2016. – 1. radionica
- 05.11.2016. – **2. radionica**
- 19.11.2016. – **1. kolo Lige programiranja**
- 03.12.2016. – 3. radionica
- 17.12.2016. – 2. kolo Lige programiranja
- termini u 2017. godini će biti naknadno određeni
- Web stranica Lige programiranja:
www.futura.com.hr/liga-programiranja-u-pythonu-2016-2017/

Ponavljjanje gradiva s 1. radionice



□ Aritmetički operatori

zbrajanje	+
oduzimanje	-
množenje	*
dijeljenje	/
cjelobrojno dijeljenje	//
modulo (ostatak od dijeljenja)	%
potenciranje	**

```
>>> a = 4
>>> b = 5
>>> a + b
9
>>> a / b
0.8
>>> a // b
0
>>> a - b * 2
-6
>>> (a - b) * 2
-2
>>> a % b
4
```

- Prvenstvo pri izvođenju ima potenciranje, pa nakon toga množenje, dijeljenje, cjelobrojno dijeljenje i modulo, te na kraju zbrajanje i oduzimanje

Ponavljjanje gradiva s 1. radionice



- Posljednji zadatak: izračun prosjeka ocjena

varijabla

```
ocjena1 = int(input('Unesi prvu ocjenu: '))
ocjena2 = int(input('Unesi drugu ocjenu: '))
ocjena3 = int(input('Unesi treću ocjenu: '))
ocjena4 = int(input('Unesi četvrtu ocjenu: '))
prosjek = (ocjena1 + ocjena2 + ocjena3 + ocjena4)/4
print('Prosjek ocjena:', prosjek)
```



prosjek.py

print - ispis
teksta i/ili
sadržaja varijabli
na ekran

int - pretvorba
niza znakova u
cijeli broj

input - unos
podataka s
tipkovnice

Relacijski operatori



veće od	>
manje od	<
veće od ili jednako	>=
manje od ili jednako	<=
jednako	==
nije jednako	!=

- Relacijski operatori uspoređuju dva operanda. Rezultat usporedbe ima vrijednosti **True** ili **False** (*Istina* ili *Laž*).

Relacijski operatori



□ Provjeriti kako operatori djeluju:

```
>>> 3 > 2
```

```
True
```

```
>>> 3 < 2
```

```
False
```

```
>>> 3 >= 2
```

```
True
```

```
>>> 3 <= 2
```

```
False
```

```
>>> 3 == 2
```

```
False
```

```
>>> 3 != 2
```

```
True
```

```
>>> a = 2
```

```
>>> b = 7
```

```
>>> b > a
```

```
True
```

```
>>> b+1 == a*4
```

```
True
```

```
>>> b/a != b//a
```

```
True
```

```
>>> (a+b)**2 <=
```

```
10*a
```

```
False
```

```
>>> r1 = 'tri'
```

```
>>> r2 = 'pet'
```

```
>>> r1 > r2
```

```
True
```

```
>>> m = 'tri'
```

```
>>> n = 'Tri'
```

```
>>> m == n
```

```
False
```

```
>>> m > n
```

```
True
```

Nizovi znakova se uspoređuju slovo po slovo (uzimajući u obzir ASCII kod)

Relacijski operatori



- Redoslijed znakova (ASCII tablica):

	!	"	#	\$	%	&	'	()	*	+	,	-	.	/
0	1	2	3	4	5	6	7	8	9	:	;	<	=	>	?
@	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
P	Q	R	S	T	U	V	W	X	Y	Z	[\]	^	_
`	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o
p	q	r	s	t	u	v	w	x	y	z	{		}	~	

Zato je:
't' > 'T'

Usput: kako
saznati ASCII
kod nekog slova?

```
>>> ord('T')  
84  
>>> ord('t')  
116
```

Donošenje odluka u programima



- Kako izvesti programsku naredbu (ili više njih) na temelju ispitivanja nekog uvjeta:

```
...  
ako je uvjet onda  
    naredba1_1  
    ...  
    naredba1_n  
...
```



Donošenje odluka u programima



□ U Pythonu:

```
...  
if uvjet:  
    → naredba1_1  
    → ...  
    → naredba1_n  
...
```

Izvršit će se
ako je *uvjet*
zadovoljen
(logički izraz
je istinit)

Svaki redak koji će se izvesti
ako je *uvjet* zadovoljen mora
biti uvučen. Najbolje je
koristiti tipku TAB



□ Primjer:

```
if a > b:  
    print ('A je veće od B')
```

Zadatak: Neparan broj

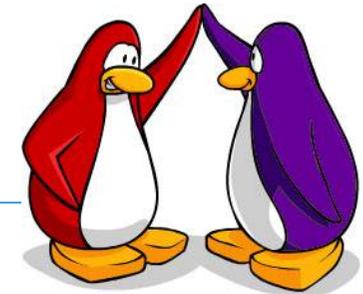


- Napisati program koji će za uneseni cijeli broj ispisati da li je taj broj neparan. Na kraju programa ispisati poruku 'Kraj programa'.

- Sjećate li se kako se piše i pokreće Python program?



Podsjetnik: Python program



The image shows a composite of three Python 3.4.1 interface windows with blue callout boxes and arrows pointing to specific elements:

- Python 3.4.1 Shell:** The top-left window shows the menu bar with 'File', 'Edit', 'Shell', 'Debug', 'Options', 'Windows', and 'Help'. The 'File' menu is open, showing options like 'New File (Ctrl+N)', 'Open...', 'Recent Files', 'Save', and 'Exit'. A blue callout box labeled 'Python IDLE' points to the window title.
- Python 3.4.1: Untitled*:** The middle window shows a code editor with the following content:

```
#Jednolinijski komentari
#Ovo je moj prvi Python program.

"""
Komentar u više linija.
Mogu se koristiti i jednostruki navodnici.
Ovo je moj prvi Python progr.
"""

'Ovo je također jednolinijski komentar.'
'Ovo je moj prvi Python program.'

print('Hello World!')
```

Annotations include:
 - 'Python IDLE editor' pointing to the window title.
 - 'komentari' pointing to the multi-line comment block.
 - 'kod programa' pointing to the `print('Hello World!')` line.
 - 'ekstenzija .py' pointing to the file extension in the 'Save As' dialog.
 - 'unos naziva programa' pointing to the filename input field.
- Save As:** The bottom-right window shows a file save dialog with 'File name: helloworld.py' and 'Save as type: Python files (*.py;*.pyw)'. A blue callout box labeled 'spremanje programa' points to the dialog.

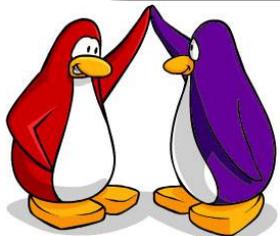
Pokretanje programa u Python IDLE-u

The image shows a screenshot of the Python IDLE environment. The main window displays a Python script named 'helloworld.py' with the following code:

```
#Jednolinijski  
#Ovo je moj pr  
  
"""  
Komentar u vi  
Mogu se koristiti i jednorazni navodnici.  
Ovo je moj prvi Python progr.  
"""  
  
'Ovo je također jednolinijski komentar.'  
"Ovo je moj prvi Python program."  
  
print('Hello World!')
```

The 'Run' menu is open, showing the 'Run Module' option (F5) highlighted. A blue callout bubble points to this option with the text 'pokretanje programa'. Another blue callout bubble points to the 'Run Module' option with the text 'prečac na tipkovnici: funkcijska tipka F5'. The 'Python 3.4.1 Shell' window shows the output of the program:

```
Python 3.4.1 (v3.4.1:c0e311e010fc, May 18 2014, 10:38:22) [MSC v.1600 32 bit (Intel)] on win32  
Type "copyright", "credits" or "license()" for more information.  
>>> ===== RESTART =====>>>  
>>> Hello World!  
>>> |
```



helloworld.py

Zadatak: Neparan broj



- Napisati program koji će za uneseni cijeli broj ispisati da li je taj broj neparan. Na kraju programa ispisati poruku 'Kraj programa'.
- Rješenje:

```
broj = int(input('Unesi broj: '))
if broj%2 != 0:
    print('Broj', broj, 'je neparan!')
print('Kraj programa')
```



neparanBroj.py

Vrijeme

Two screenshots of a Python 3.5.0 Shell window. The left screenshot shows the program being executed with the input '11', resulting in the output 'Broj 11 je neparan!' and 'Kraj programa'. The right screenshot shows the program being executed with the input '6', resulting in the output 'Kraj programa'. Both screenshots show the file path: D:/Dropbox/futura-radionice/Liga_programiranja_2016/02radionica/02-primjeri_zadaci/neparanBroj.py.

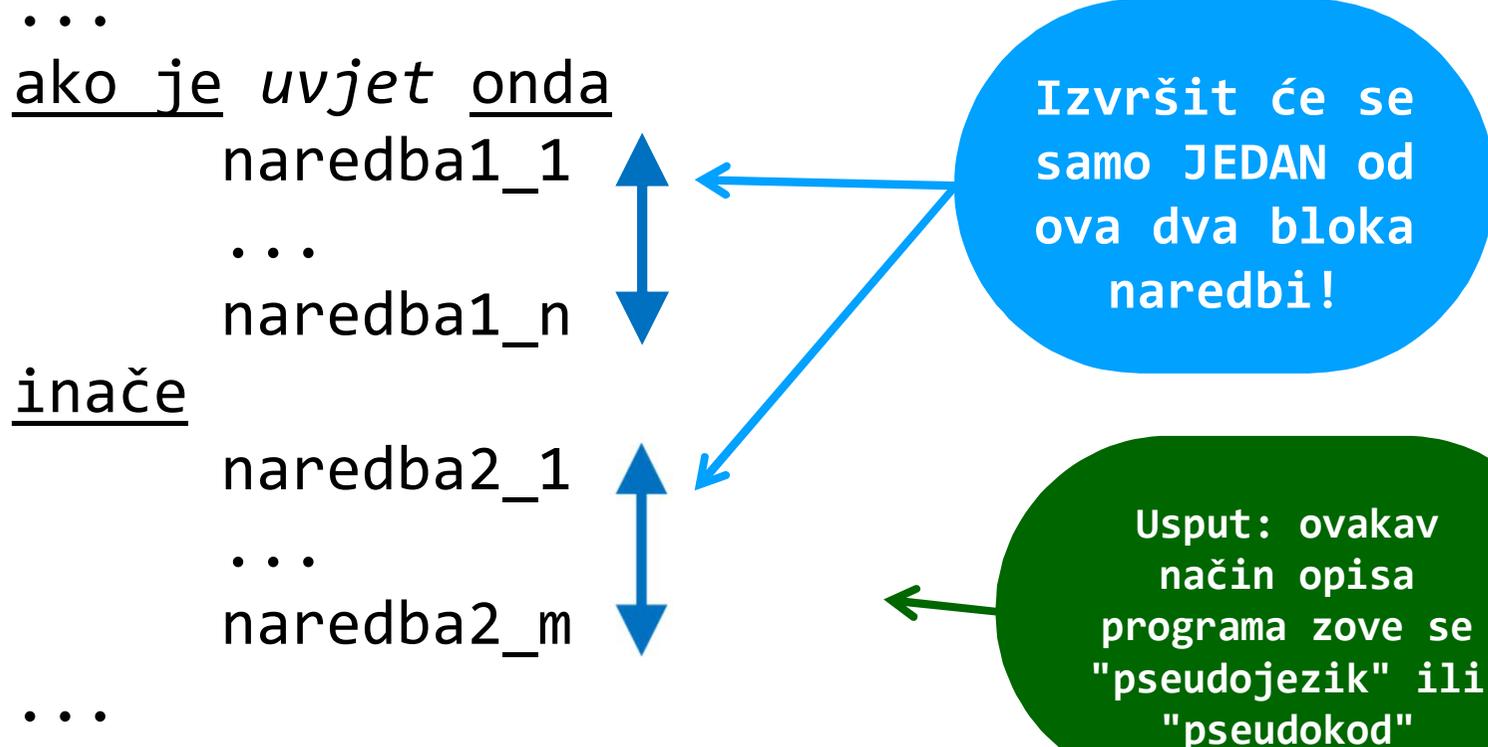
```
Python 3.5.0 Shell
File Edit Shell Debug Options Window Help
Python 3.5.0 (v3.5.0:374f501f4567, Sep 13 2015)
Type "copyright", "credits" or "license()" for more
>>>
RESTART: D:/Dropbox/futura-radionice/Liga_programiranja_2016/02radionica/02-primjeri_zadaci/neparanBroj.py
Unesi broj: 11
Broj 11 je neparan!
Kraj programa
>>>

Python 3.5.0 Shell
File Edit Shell Debug Options Window Help
>>>
RESTART: D:/Dropbox/futura-radionice/Liga_programiranja_2016/02radionica/02-primjeri_zadaci/neparanBroj.py
Unesi broj: 6
Kraj programa
>>>
Ln: 11 Col: 0
```

Donošenje odluka u programima



- Često je u programima potrebno odabrati jednu od dvije mogućnosti:



Donošenje odluka u programima



□ U Pythonu:

```
...  
if uvjet:  
    naredba1_1  
    ...  
    naredba1_n  
else:  
    naredba2_1  
    ...  
    naredba2_m  
...
```



Izvršit će se samo JEDAN od ova dva bloka naredbi!

Donošenje odluka u programima



□ U Pythonu:

```
...  
if uvjet:  
    naredba1_1  
    ...  
    naredba1_n  
  
else:  
    naredba2_1  
    ...  
    naredba2_m  
  
...
```



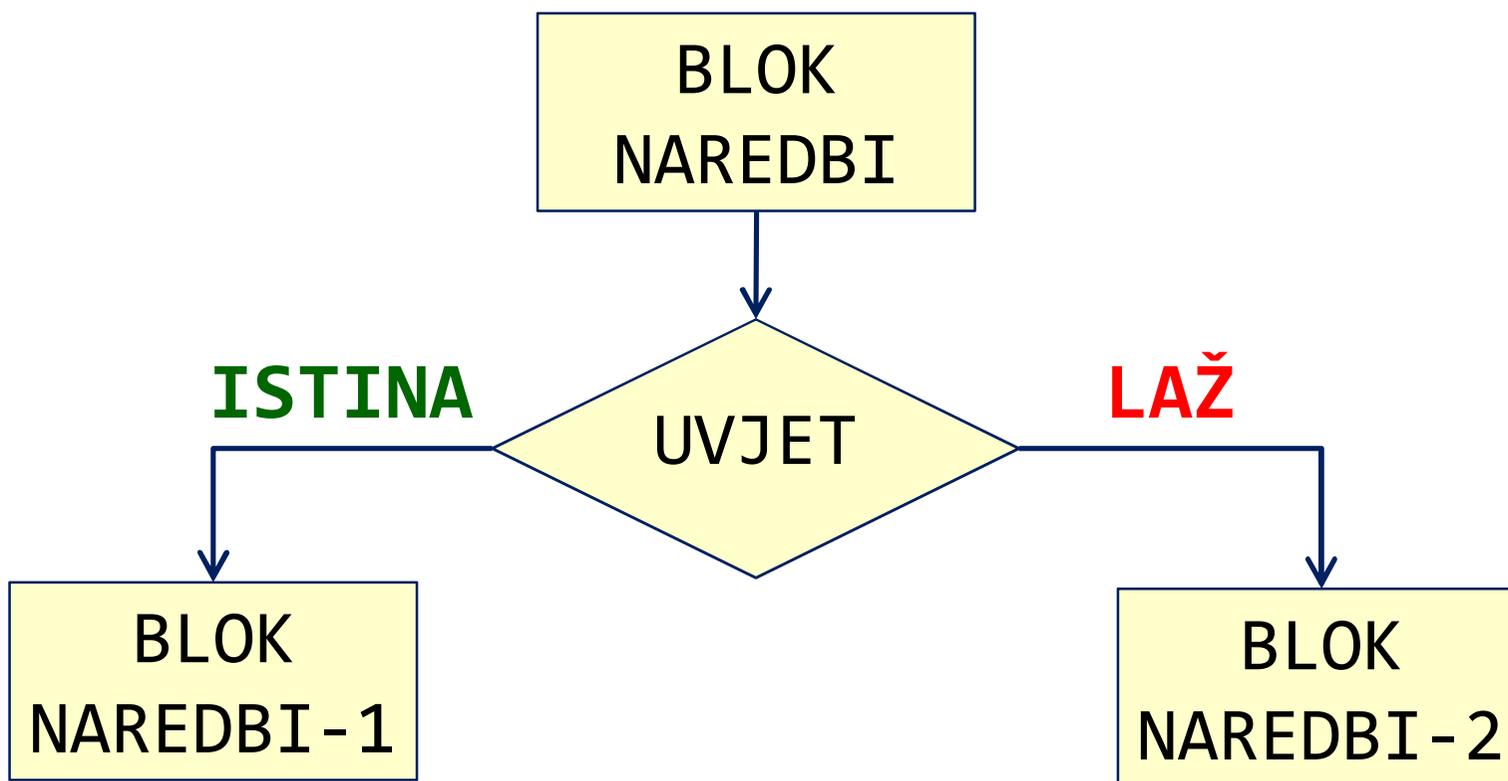
Izvršit će se
ako je *uvjet*
zadovoljen
(logički izraz
je istinit)

Izvršit će se
ako *uvjet* nije
zadovoljen
(logički izraz
je lažan)

Donošenje odluka u programima



- Dakle, na temelju ispitivanja *uvjeta* imamo grananje programa:



Donošenje odluka u programima

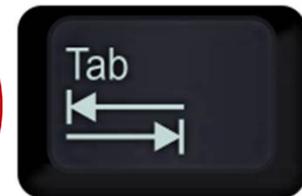


- U Pythonu je važno paziti na uvlačenje redaka naredbi koje su dio bloka :

```
...  
if uvjet:  
    → naredba1_1  
    → ...  
    → naredba1_n  
else:  
    → naredba2_1  
    → ...  
    → naredba2_m  
...
```

Izvršit će se samo JEDAN od ova dva bloka naredbi!

Najbolje je koristiti tipku TAB



Zadatak: Čokolada

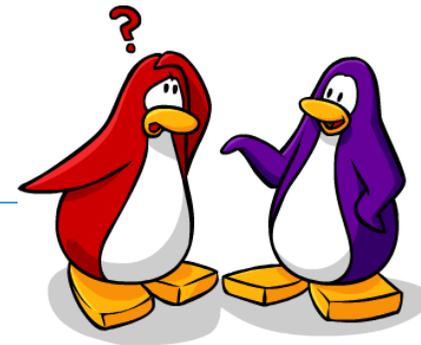


- Marko želi kupiti čokoladu koja ima toliko kockica da se može ravnomjerno i bez ostatka podijeliti na **N** prijatelja.
- Napiši program u koji se unosi broj redaka čokolade **R**, broj stupaca čokolade **S** i broj prijatelja **N**. Program treba ispisati da li se sve kockice čokolade mogu podijeliti ravnomjerno i koliko kockica dobije svaki prijatelj ako se mogu podijeliti.
- Ulazni podaci: tri cijela broja, svaki u svom retku
 - Cijeli broj R ($1 \leq R \leq 100$), broj redaka čokolade
 - Cijeli broj S ($1 \leq S \leq 100$), broj stupaca čokolade
 - Cijeli broj N ($1 \leq N \leq 100$), broj prijatelja
- Izlazni podaci:
 - Poruka 'Čokolada se može podijeliti' ili 'Čokolada se ne ravnopravno može podijeliti'
 - Cijeli broj (broj kockica koje dobije svaki prijatelj) ili 0 ako ne može

Primjeri testnih podataka

ULAZ	ULAZ
5	4
3	6
5	7
IZLAZ	IZLAZ
Čokolada se može podijeliti	Čokolada se ne može ravnopravno podijeliti
3	0

Kakav je ovo zadatak!?!



- ❑ Zadaci na natjecanjima najčešće su zadani u obliku priče koja bi sudionicima natjecanja trebala biti zabavna, zanimljiva i poticajna
- ❑ Priča opisuje problemsko područje i postavlja okvir za rješavanja zadatka
- ❑ Naglašeni su specifični uvjeti i ograničenja
- ❑ Detaljno je opisan oblik ulaznih podataka i izlaznih rezultata
- ❑ Obavezno je navedeno nekoliko ulaznih podataka i očekivanih izlaza
 - **OPREZ!** Navedeni podaci za testiranje često ne pokrivaju SVE moguće situacije

Pristup rješenju



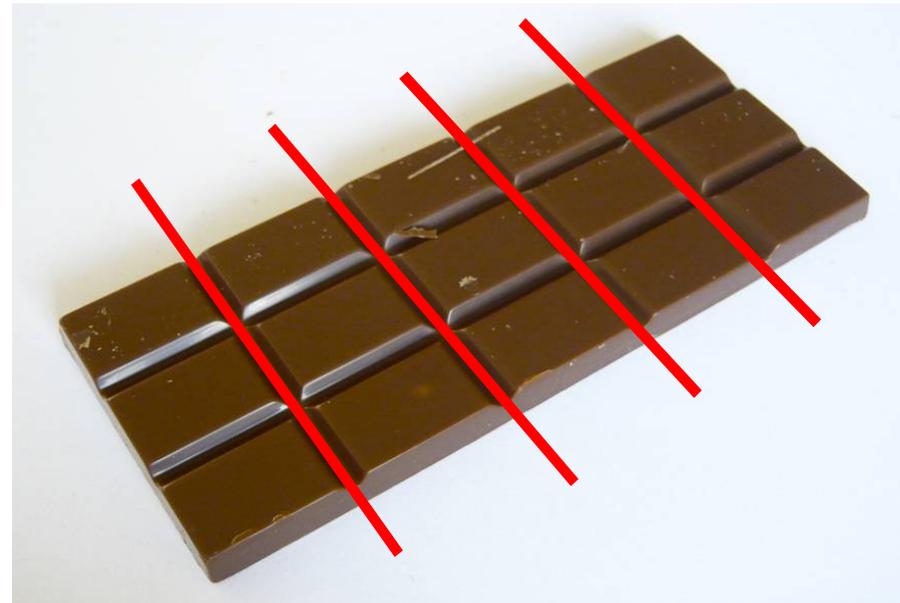
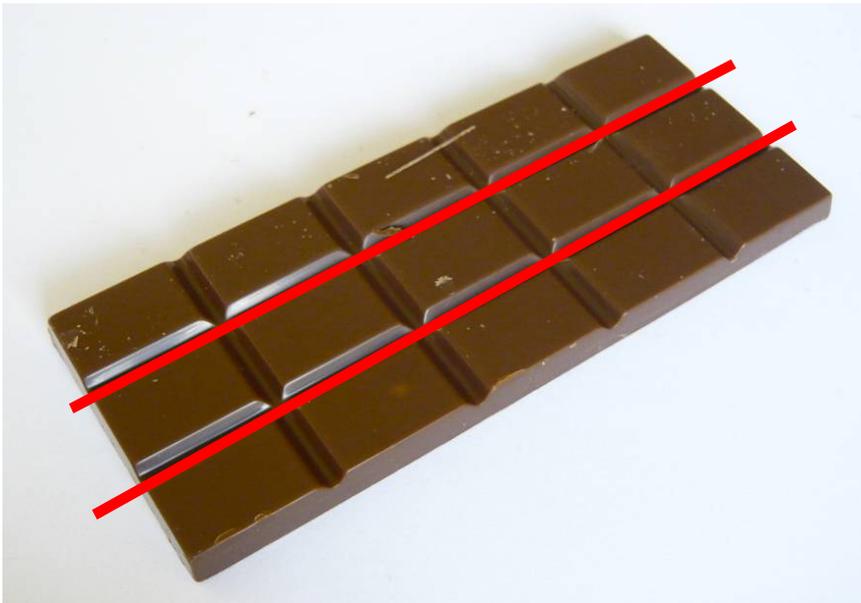
- 1. Pročitaj zadatak i shvati ga
 - Ne idi dalje dok nisi shvatio zadatak!**
2. Skiciraj rješenje
 - Blok dijagramom,
 - Pseudokodom,
 - Slobodnim tekstom,...
3. Izaberi alat (programski jezik)
4. Programiraj u **malim koracima**
 - 1. Isprogramiraj mali komadić koda
 2. Istestiraj napisani komadić koda
 3. Ako je do tada napisani kod u redu, dodaj novi



Zadatak: Čokolada



- Primjer: Čokolada ima 15 kockica
 - može se podijeliti na 3 prijatelja
 - može se podijeliti na 5 prijatelja



Zadatak: Čokolada

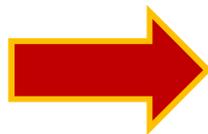


- Marko želi kupiti čokoladu koja ima toliko kockica da se može ravnomjerno i bez ostatka podijeliti na **N** prijatelja.
- Napiši program u koji se unosi broj redaka čokolade **R**, broj stupaca čokolade **S** i broj prijatelja **N**. Program treba ispisati da li se sve kockice čokolade mogu podijeliti ravnomjerno i koliko kockica dobije svaki prijatelj ako se mogu podijeliti.
- Ulazni podaci: tri cijela broja, svaki u svom retku
 - Cijeli broj R ($1 \leq R \leq 100$), broj redaka čokolade
 - Cijeli broj S ($1 \leq S \leq 100$), broj stupaca čokolade
 - Cijeli broj N ($1 \leq N \leq 100$), broj prijatelja
- Izlazni podaci:
 - Poruka 'Čokolada se može podijeliti' ili 'Čokolada se ne ravnopravno može podijeliti'
 - Cijeli broj (broj kockica koje dobije svaki prijatelj) ili 0 ako ne može

Primjeri testnih podataka

ULAZ	ULAZ
5	4
3	6
5	7
IZLAZ	IZLAZ
Čokolada se može podijeliti	Čokolada se ne može ravnopravno podijeliti
3	0

Vrijeme



Zadatak: Čokolada - rješenje



```
redovi = int(input('Unesi broj redova: '))
stupci = int(input('Unesi broj stupaca: '))
prijatelji = int(input('Unesi broj prijatelja: '))
if (redovi*stupci) % prijatelji == 0:
    print('Čokolada se može podijeliti')
    print((redovi*stupci) // prijatelji)
else:
    print('Čokolada se ne može ravnopravno podijeliti')
    print(0)
```



cokolada.py

Two screenshots of a Python 3.5.0 Shell window. The left screenshot shows the program being executed with input values 5, 3, and 5, resulting in the output 'Čokolada se može podijeliti' and the number 3. The right screenshot shows the program being executed with input values 4, 6, and 7, resulting in the output 'Čokolada se ne može ravnopravno podijeliti' and the number 0. Both screenshots show the file path 'D:/Dropbox/futura-radionice/Liga_programiranja_2016/02radionica/02-prijatelj_zadaci/cokolada.py' and the Python version 'Python 3.5.0 (v3.5.0:374f501f4567, Sep 13 2015, 02:16:59) [MSC v.1900 32 bit (Intel)] on win32'.

```
Python 3.5.0 Shell
File Edit Shell Debug Options Window Help
Python 3.5.0 (v3.5.0:374f501f4567, Sep 13 2015, 02:16:59) [MSC v.1900 32 bit (Intel)] on win32
Type "copyright", "credits" or "license()" for more
>>>
RESTART: D:/Dropbox/futura-radionice/Liga_programiranja_2016/02radionica/02-prijatelj_zadaci/cokolada.py
Unesi broj redova: 5
Unesi broj stupaca: 3
Unesi broj prijatelja: 5
Čokolada se može podijeliti
3
>>> |

Python 3.5.0 Shell
File Edit Shell Debug Options Window Help
Python 3.5.0 (v3.5.0:374f501f4567, Sep 13 2015, 02:16:59) [MSC v.1900 32 bit (Intel)] on win32
Type "copyright", "credits" or "license()" for more information.
>>>
RESTART: D:/Dropbox/futura-radionice/Liga_programiranja_2016/02radionica/02-prijatelj_zadaci/cokolada.py
Unesi broj redova: 4
Unesi broj stupaca: 6
Unesi broj prijatelja: 7
Čokolada se ne može ravnopravno podijeliti
0
>>> |

Ln: 10 Col: 4
```

Donošenje odluka u programima



- U slučaju višestrukog izbora:

```
...  
if uvjet_1:  
    blok naredbi_1  
elif uvjet_2:  
    blok naredbi_2  
...  
elif uvjet_n:  
    blok naredbi_n  
else:  
    blok naredbi  
...
```

Izvršit će se
samo JEDAN od
blokova
naredbi!

Zadatak: Nogomet



- Đive i Maro se ne mogu dogovoriti tko je pobjednik na nogometnoj utakmici plavih protiv bijelih, pa to treba riješiti programom.
- Potrebno je unijeti broj postignutih golova plavih i bijelih, i onda ispisati tko je pobjednik i s koliko golova razlike, ili poruku da nema pobjednika.
- Ulazni podaci: dva cijela broja, svaki u svom retku
 - Cijeli broj PLAVI ($1 \leq \text{PLAVI} \leq 30$), broj golova plavih
 - Cijeli broj BIJELI ($1 \leq \text{BIJELI} \leq 30$), broj golova bijelih
- Izlazni podaci:
 - Poruka tko je pobjednik i s koliko golova razlike, ili da nema pobjednika

Vrijeme



Primjeri testnih podataka

ULAZ	ULAZ	ULAZ
3	3	0
2	5	0
IZLAZ	IZLAZ	IZLAZ
Pobijedili su plavi, gol razlika plavih: 1	Pobijedili su bijeli, gol razlika bijelih: 2	Nema pobjednika

Zadatak: Nogomet - rješenje



```
plavi = int(input('Unesi broj golova plavih: '))
bijeli = int(input('Unesi broj golova bijelih: '))
if plavi > bijeli:
    print('Pobijedili su plavi, gol razlika plavih:',
          plavi - bijeli)
elif bijeli > plavi:
    print('Pobijedili su bijeli, gol razlika bijeli:',
          bijeli - plavi)
else:
    print('Nema pobjednika')
```



nogomet.py

<pre>Python 3.5.0 Shell File Edit Shell Debug Options Window Help Python 3.5.0 (v3.5.0:374f501f4567, Sep 13 2015) on win32 Type "copyright", "credits" or "license()" for more >>> RESTART: D:/Dropbox/futura-radionice/Liga_mjeri_zadaci/nogomet.py Unesi broj golova plavih: 3 Unesi broj golova bijelih: 2 Pobijedili su plavi, gol razlika plavih: 1 >>></pre>	<pre>Python 3.5.0 Shell File Edit Shell Debug Options Window Help Python 3.5.0 (v3.5.0:374f501f4567, Sep 13 2015) on win32 Type "copyright", "credits" or "license()" for more >>> RESTART: D:/Dropbox/futura-radionice/Liga_mjeri_zadaci/nogomet.py Unesi broj golova plavih: 3 Unesi broj golova bijelih: 5 Pobijedili su bijeli, gol razlika bijeli: 2 >>> </pre>	<pre>Python 3.5.0 Shell File Edit Shell Debug Options Window Help Python 3.5.0 (v3.5.0:374f501f4567, Sep 13 2015) on win32 Type "copyright", "credits" or "license()" for more >>> RESTART: D:/Dropbox/futura-radionice/Liga_mjeri_zadaci/nogomet.py Unesi broj golova plavih: 0 Unesi broj golova bijelih: 0 Nema pobjednika >>> </pre>
--	---	---

Logički operatori i logički izrazi



- Što ako je uvjet na temelju kojeg treba donijeti odluku složen?
- Logički operatori:

logička I operacija	and
logička ILI operacija	or
NE operacija (negacija)	not

- Redoslijed izvođenja logičkih operacija:

1.	not
2.	and
3.	or

Logički operatori i logički izrazi



□ Primjeri logičkih operacija:

```
>>> a = 2
>>> b = 3
>>> c = 10
>>> a > b
False
>>> c > b
True
>>> a > b and c > b
False
>>> a > b or c > b
True
```

Za logičku **AND** operaciju rezultat će biti **True** (istina) samo ako su oba izraza True (istinita)

Za logičku **OR** operaciju rezultat će biti **True** (istina) već ako je jedan od izraza True (istinit)

Logički operatori i logički izrazi



- Kako se podsjetiti ishoda logičkih operacija?

and (i)

```
>>> False and False
False
>>> False and True
False
>>> True and False
False
>>> True and True
True
```

or (ili)

```
>>> False or False
False
>>> False or True
True
>>> True or False
True
>>> True or True
True
```

not (ne)

```
>>> not False
True
>>> not True
False
```

- Je li isto **True** i **true**? Ili **False** i **false**?
- Što je rezultat izraza: **True and true**

Logički operatori i logički izrazi



□ Primjeri logičkih operacija:

```
>>> a = 2
>>> b = 3
>>> c = 10
>>> a+2*3>=c or not(a>b) and a*b-2==c%6
```

1.	aritmetički
2.	relacijski
3.	logički

Ako imamo kombinirane aritmetičke, relacijske i logičke operatore, onda je ovo redosljed izvođenja operacija.

Logički operatori i logički izrazi



□ Primjeri logičkih operacija:

```
>>> a = 2
>>> b = 3
>>> c = 10
>>> a+2*3>=c or not(a>b) and a*b-2==c%6
True
>>>
>>> (a+2*3>=c) or (not(a>b) and (a*b-2==c%6))
True
>>>
```

Ako ipak nismo posve sigurni
u redosljed operacija onda
je najbolje koristiti
zagrade!

Zadatak: Ispit



- Napisati program koji će na temelju unesenog broja bodova (na jedno decimalno mjesto) ispisati ocjenu ispita. Na ispitu se može ostvariti najmanje 0, a najviše 100 bodova.

Nedovoljan => bodovi ≥ 0.0 i bodovi < 50.0

Dovoljan => bodovi ≥ 50.0 i bodovi < 60.5

Dobar => bodovi ≥ 60.5 i bodovi < 75.5

Vrlo dobar => bodovi ≥ 75.5 i bodovi < 85.0

Odličan => bodovi ≥ 85.0 i bodovi ≤ 100.0

- Ako broj bodova nije među navedenim vrijednostima ispisati grešku!

Vrijeme



Zadatak: Ispit - rješenje



```
bodovi = float(input('Unesi broj bodova: '))
if bodovi >= 0.0 and bodovi < 50.0:
    print('Nedovoljan')
elif bodovi >= 50.0 and bodovi < 60.5:
    print('Dovoljan')
elif bodovi >= 60.5 and bodovi < 75.5:
    print('Dobar')
elif bodovi >= 75.5 and bodovi < 85.0:
    print('Vrlo dobar')
elif bodovi >= 85.0 and bodovi <= 100.0:
    print('Odličan')
else:
    print('Unesen je pogrešan broj bodova!')
```



ispit.py

Testirati program s dovoljno vrijednosti – npr.:

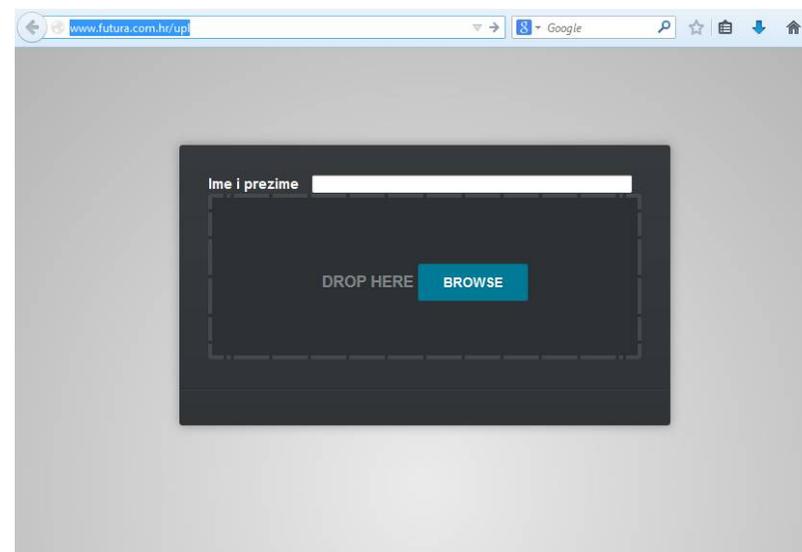
7.5 75.5 90 60.4 89.4 111.1 60.5

Slanje programa na natjecanju



- Kad idući put bude kolo Lige programiranja, bit će potrebno poslati (upload) programski kôd riješenih ili djelomično riješenih zadataka.
- Link za slanje programa:

www.futura.com.hr/upl



Slanje programa na natjecanju



1. Upisati ime

2. Za svaki program:
- „drag & drop”
ili
- koristiti tipku „browse”

Ne zaboravite!

- Za 15 dana – u subotu 19.11.2016. –
1. kolo Lige programiranja
- 5./6. razredi** - početak **9:00**
- 7./8. razredi** - početak **10:30**
- 3 zadatka rješavate 75 minuta**
- nemojte kasniti!**

