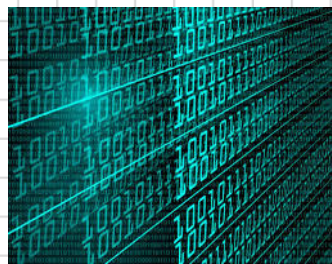




NAČRT LEKCIJE



PROGRAMIRANJE IN KODIRANJE V POKLICNEM IZOBRAŽEVANJU

2023-2-SK01-KA210-VET-000184496 01. 01. 2024 – 31. 12. 2025

CNC

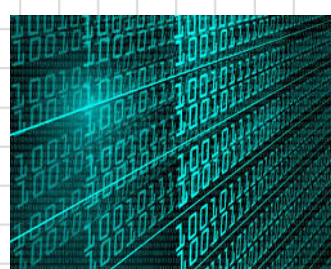
Financira Evropska unija. Izražena stališča in mnenja so le avtorjev in ne odražajo nujno stališč Evropske unije ali Evropskega izobraževalnega in slovaškega akademskega združenja za mednarodno sodelovanje (SAAIC). Niti Evropska unija niti SAAIC ne moreta biti odgovorna zanje.



Funded by
the European Union



UČNA PRIPRAVA



ZADEVA: PROGRAMIRANJE

TEMA LEKCIJE: PROGRAMIRANJE V PYTHONU – RISANJE GEOMETRIJSKIH LIKOV.

UČNI CILJI: 1. PRIDOBI OSNOVNO ZNANJE PROGRAMIRANJA V PYTHONU Z UPORABO GRAFIČNE KNJIŽNICE TKINTER.

2. NAUČITI SE DELATI Z GRAFIČNIM PLATNOM. NANJ NARIŠITE PREPROSTE OBLIKE.

3. USTVARITE SLIKE NA NAKLJUČNI LOKACIJI S KNJIŽNICO TKINTER.

ORODJA:

**RAČUNALNIK Z NAMEŠČENIM
PYTHONOM IN KNJIŽNICO TKINTER.
PYTHON VZORČNI UČBENIŠKI DELOVNI
LIST Z VAJAMI**

**HOOK/INTRO: POKAZALI BOMO, KAKO
USTVARIMO IN POŽENEMO PRVI
PROGRAM, IZDELAMO GRAFIČNO
PLATNO, NA KATEREGA BOMO RISALI
GRAFIČNE OBLIKE. UPORABILI BOMO
SPREMENLJIVKO, USTVARILI
NAKLJUČNE VREDNOSTI IN NARISALI
SLIKE NA NAKLJUČNI LOKACIJI Z
UPORABO KNJIŽNICE TKINTER.**

**POTEK UČNE URE: UVOD (4 MINUTE) – RISANJE ELEMENTARNE SLIKE V KARTEZIČNIH
KOORDINATAH INTERVJU Z UČENCI O KOORDINATNEM SISTEMU V MODULU TKINTER (1. NALOGA)**

- **RAZISKOVANJE (12 MINUT) – RAZISKOVANJE NOVEGA UKAZA (2. DO 4. NALOGA)**
- **PREDSTAVITEV (15 MINUT) – POVZETEK IN RAZLAGA ZNANJA IZ SKLOPA RAZISKOVANJE
PROBLEMATIKE IZDELAVE PROGRAMOV Z GRAFIČNIM UPORABNIŠKIM VMESNIKOM.**
- **RAZVOJ (9 MINUT) – SAMOSTOJNO PROGRAMIRANJE ZAHTEVNEJŠIH NALOG (NALOGE 5 DO 6)**
- **OCENJEVANJE (5 MINUT) – REŠEVANJE TESTA ZA SAMOOCENJEVANJE, POGOVOR O ODGOVORIH.**

KRATEK PREGLED PROGRAMIRANJA PYTHON (ZA ZAČETNIKE).

NAMESTITEV KNJIŽNICE TKINTER (ČE NI NAMEŠČENA).

PREDSTAVITEV KNJIŽNICE TKINTER.

RAZLAGA PRINCIPA RISANJA NA PLATNO.

DELOVNI LIST ZA UČENCA

**NADALJEVALI BOMO PO NAŠEM ELEKTRONSKEM UČBENIKU PROGRAMIRANJE V PYTHONU, KI JE
NAMENJEN SREDNJIM ŠOLAM. VZOREC UČBENIKA NAJDETE NA:**

**[HTTP://WWW.PROGRAMUJEMEVPYTHONE.SK/UKAZKA.PDF](http://www.programujemevpythone.sk/ukazka.pdf) VZOREC VSEBUJE PREPROSTE PRIMERE
RISANJA:**

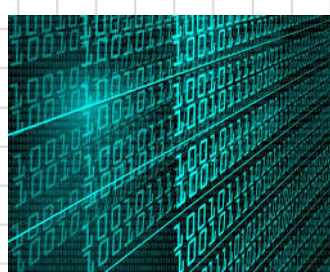
1. RISANJE OSNOVNIH OBLIK:

**RISANJE KROGA, PRAVOKOTNIKA, KVADRATA IN TRIKOTNIKA Z UPORABO RAZLIČNIH BARV IN
DEBELIN ČRT PRIMER KODE ZA VSAKO OBLIKO FAZA OSVETLITVE – OSVETLITEV NOVEGA
PREDMETA**

– NA TABLI BOM NA KRATKO ZABELEŽIL OSNOVNE POJME



UČNA PRIPRAVA

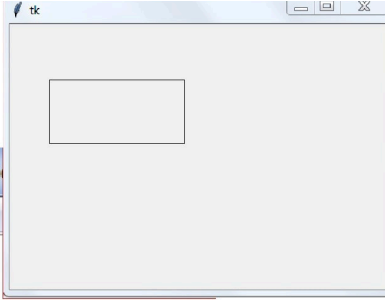


DELOVNI LIST ZA UČENCA

1. Naloga: Ustvarite datoteko 05lek_p04.py, v njej navedene ukaze in zaženite program xxxxx:

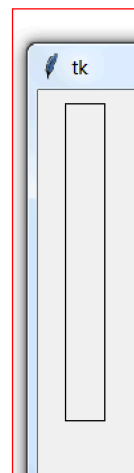
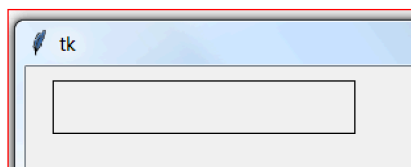
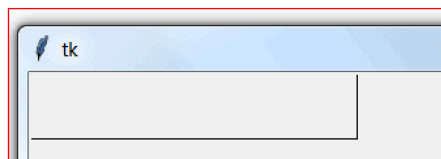
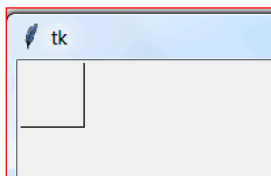
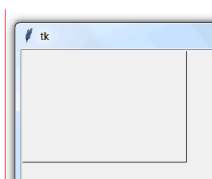
```
05lek_p04.py - F:/14_Python(SŠ)/PythonSŠ_PracovnéListy/05_PL (ri
File Edit Format Run Options Window Help
import tkinter
canvas = tkinter.Canvas()
canvas.pack()

canvas.create_rectangle(50, 70, 220, 150)
```



2. Naloga: V oklepaju ukaza `canvas.create_rectangle(,,,)` so 4 številke. Poskusite jih postopoma spremeniti v programu 05lek_p05.py (spremembe glede na prejšnji vnos so označene z rumeno). Vsakič zaženite program, da vidite, kaj nariše:

- a) `canvas.create_rectangle(0, 0, 220, 150)`
- b) `canvas.create_rectangle(0, 0, 50, 50)`
- c) `canvas.create_rectangle(0, 0, 250, 50)`
- d) `canvas.create_rectangle(20, 10, 250, 50)`
- e) `canvas.create_rectangle(20, 10, 50, 250)`





UČNA PRIPRAVA



1.

DELOVNI LIST ZA UČENCA

1. 3. Naloga: Na računalniku je v zgornjem levem kotu točka s koordinatami $[0, 0]$. Os x gre od leve proti desni. Y-os je v primerjavi z matematiko obrnjena – gre od zgoraj navzdol (večje kot je število, nižje).

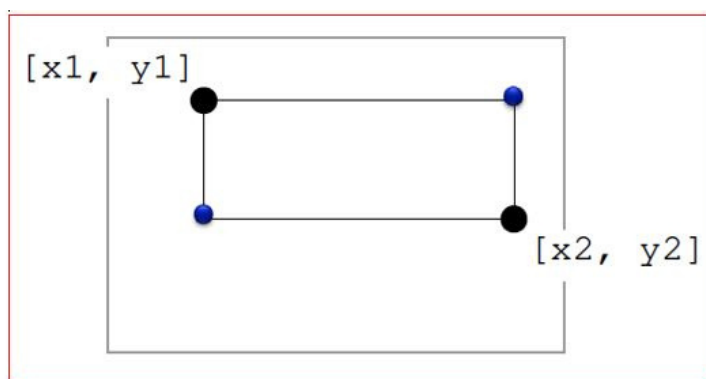


1.

4. Naloga: V ukazu `canvas.create_rectangle(x1, y1, x2, y2)`

2.

v oklepaj zapišemo koordinate nasprotnih oglišč pravokotnika:



1. 5. Naloga: Spremenite svoj program 05lek_p06.py, da nariše pravokotnik, ki ima koordinate nasprotnih oglišč $[50, 30]$ in $[300, 200]$.

```
*05lek_p05.py - F:/14_Python(SŠ)/PythonSŠ_PracovnéListy/05_PL (riešenia)/05lek_p05.py
File Edit Format Run Options Window Help
import tkinter
canvas = tkinter.Canvas()
canvas.pack()

canvas.create_rectangle(50, 30, 300, 200)
```

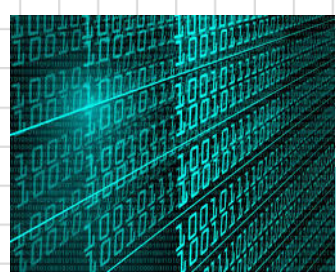
1.

6. Naloga: Brez uporabe računalnika izračunajte širino in višino pravokotnika iz pr
Širka: $300 - 50 = 250$
Výška: $200 - 30 = 170$

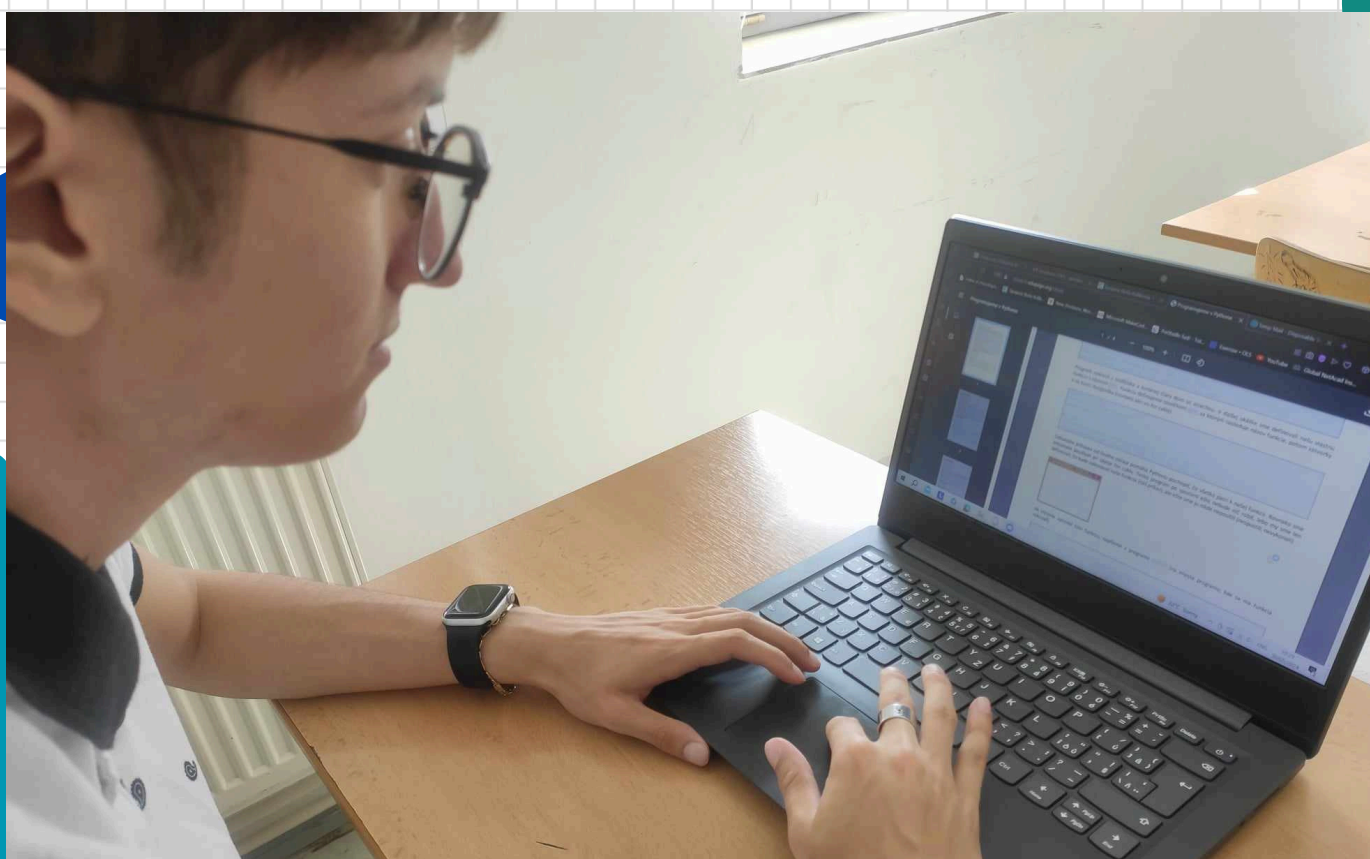
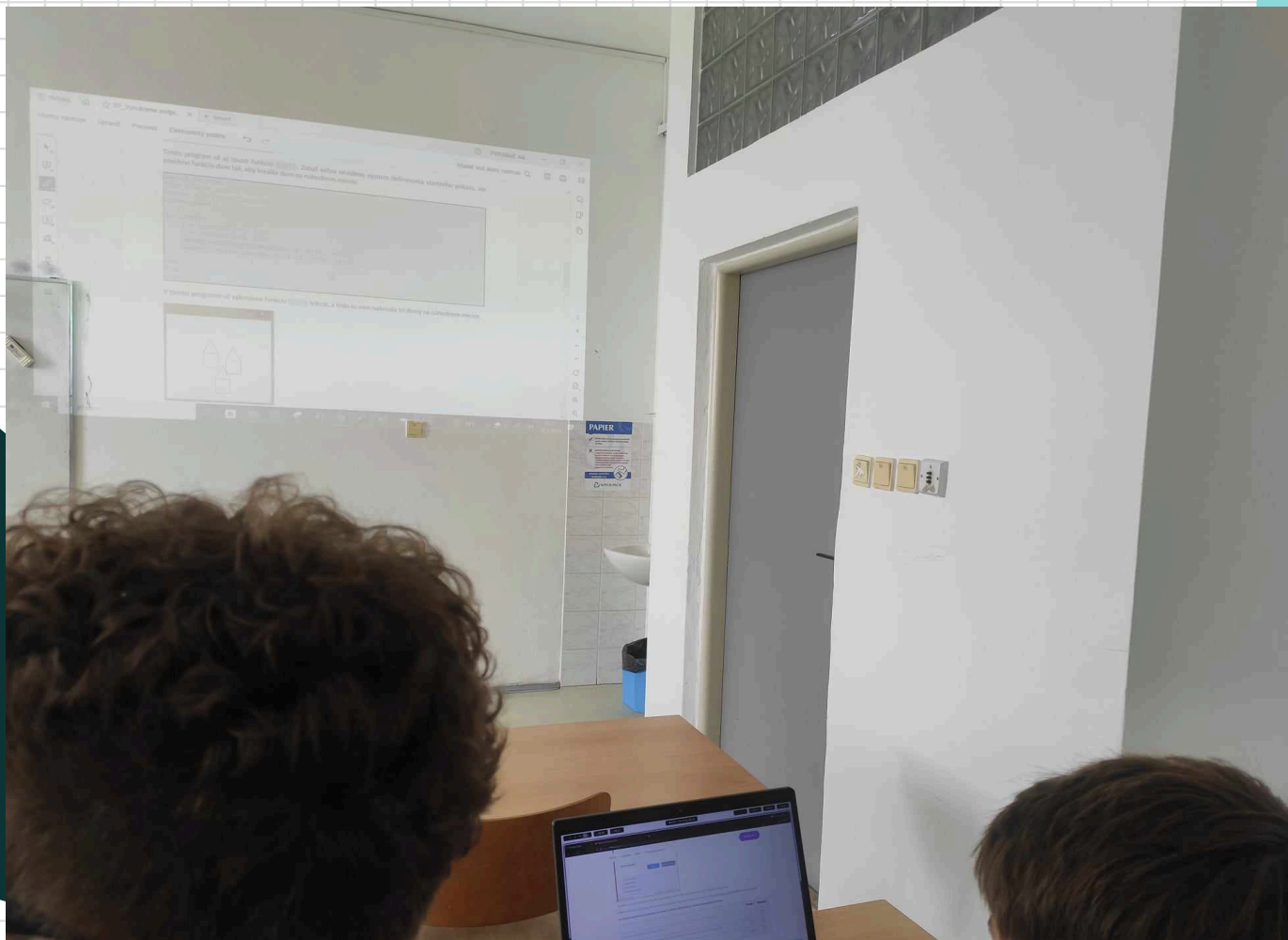
1. 7. Naloga: . Ustvarite nov program 05lek_p08.py in narišite pravokotnik, ki ima eno oglišče na koordinatah $[200, 100]$, njegova širina je 60 in višina 140.



PRILOGA

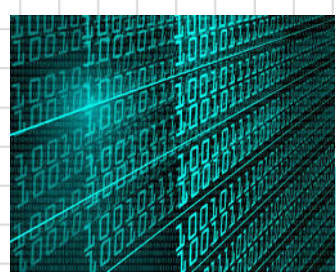


FOTOGRAFIJE IZ IZVEDBE LEKCIJE

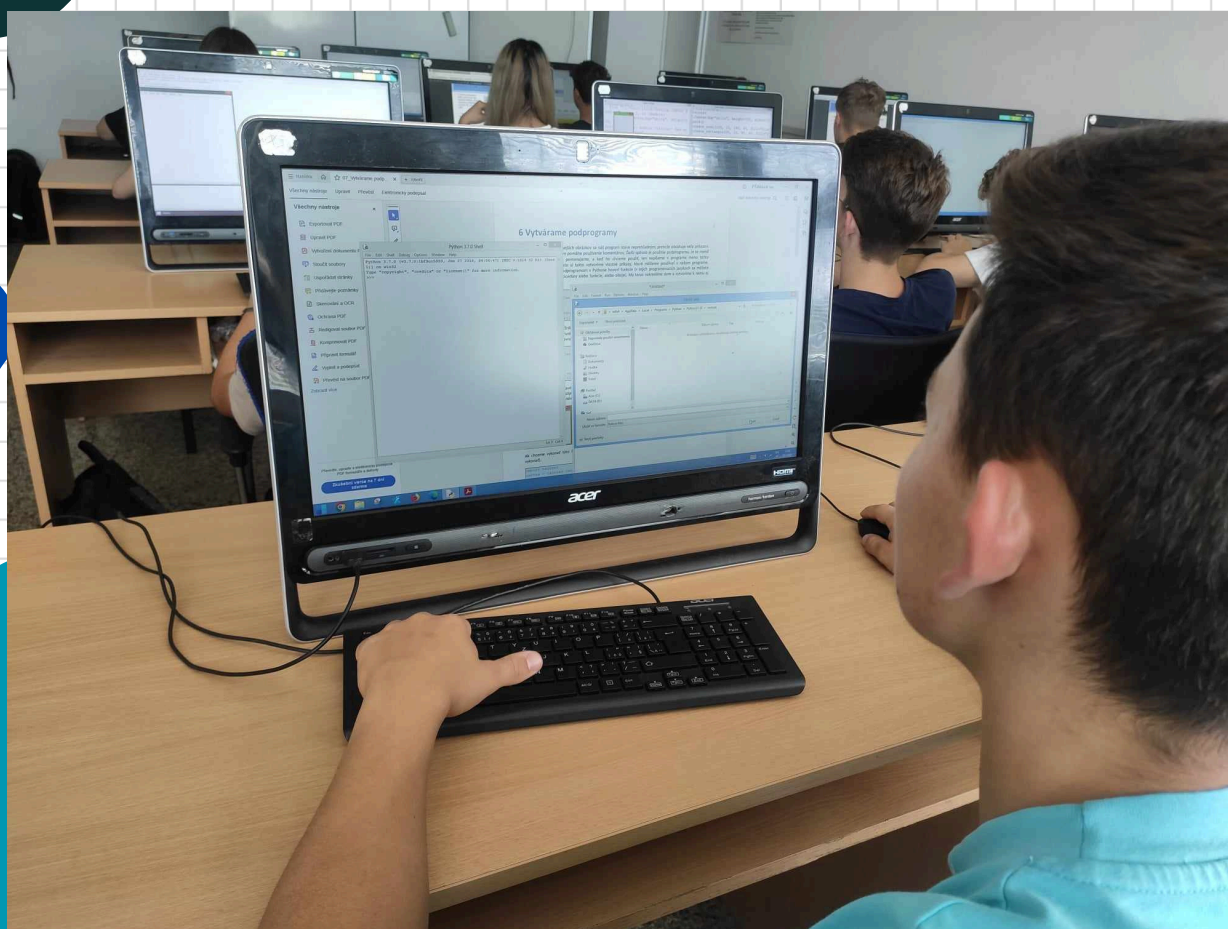
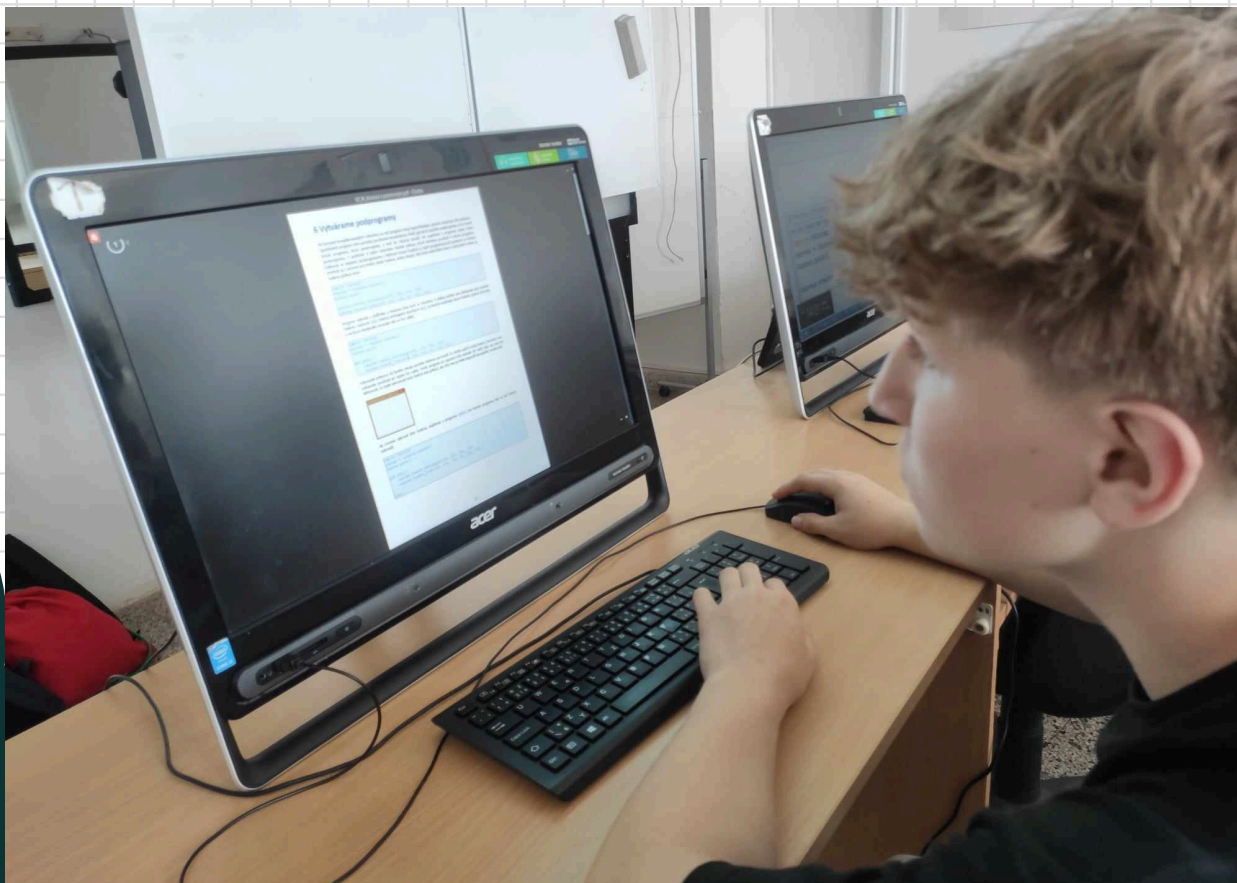




PRILOGA

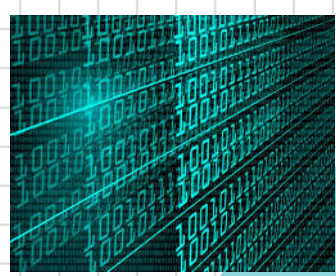


FOTOGRAFIJE IZ IZVEDBE LEKCIJE

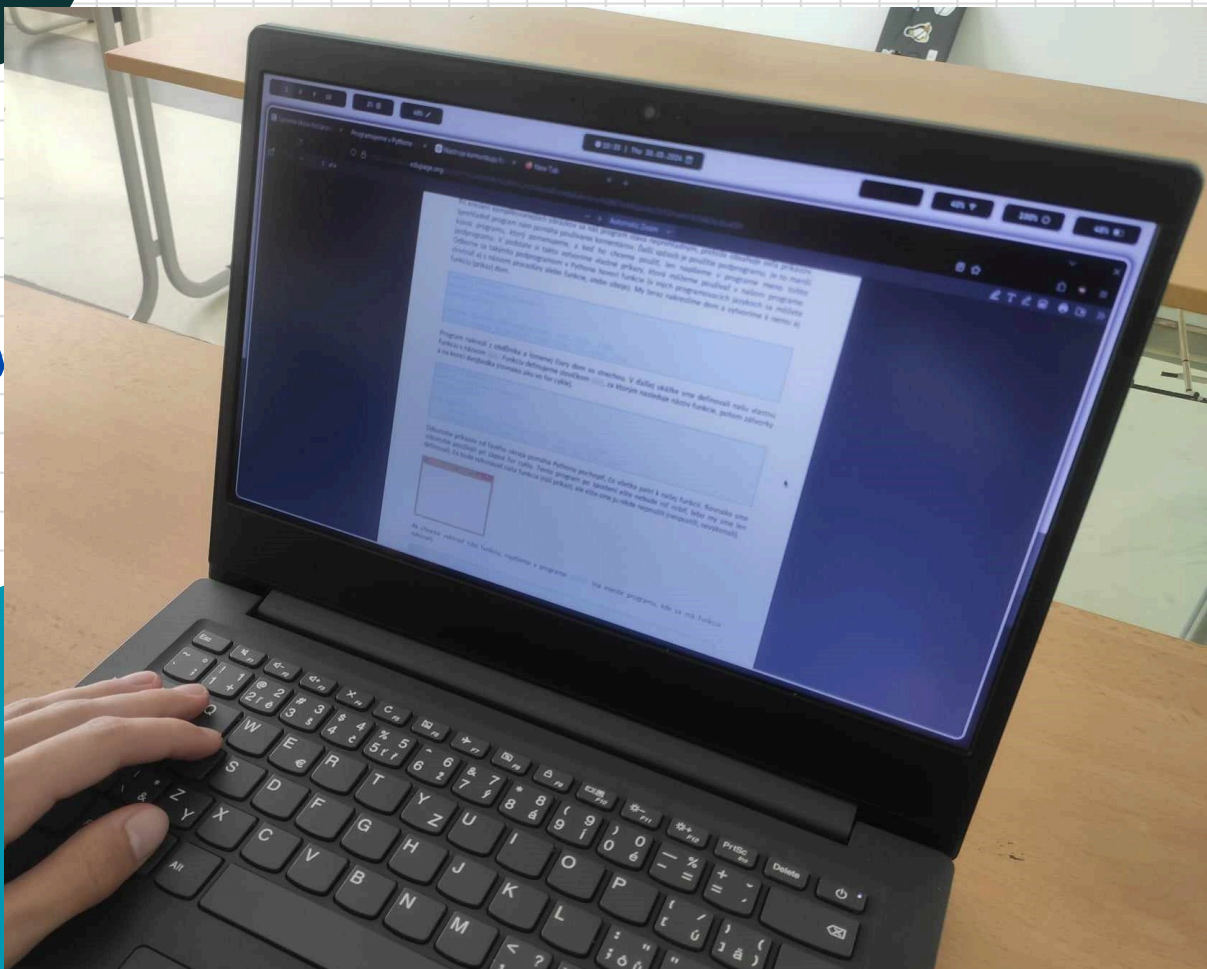
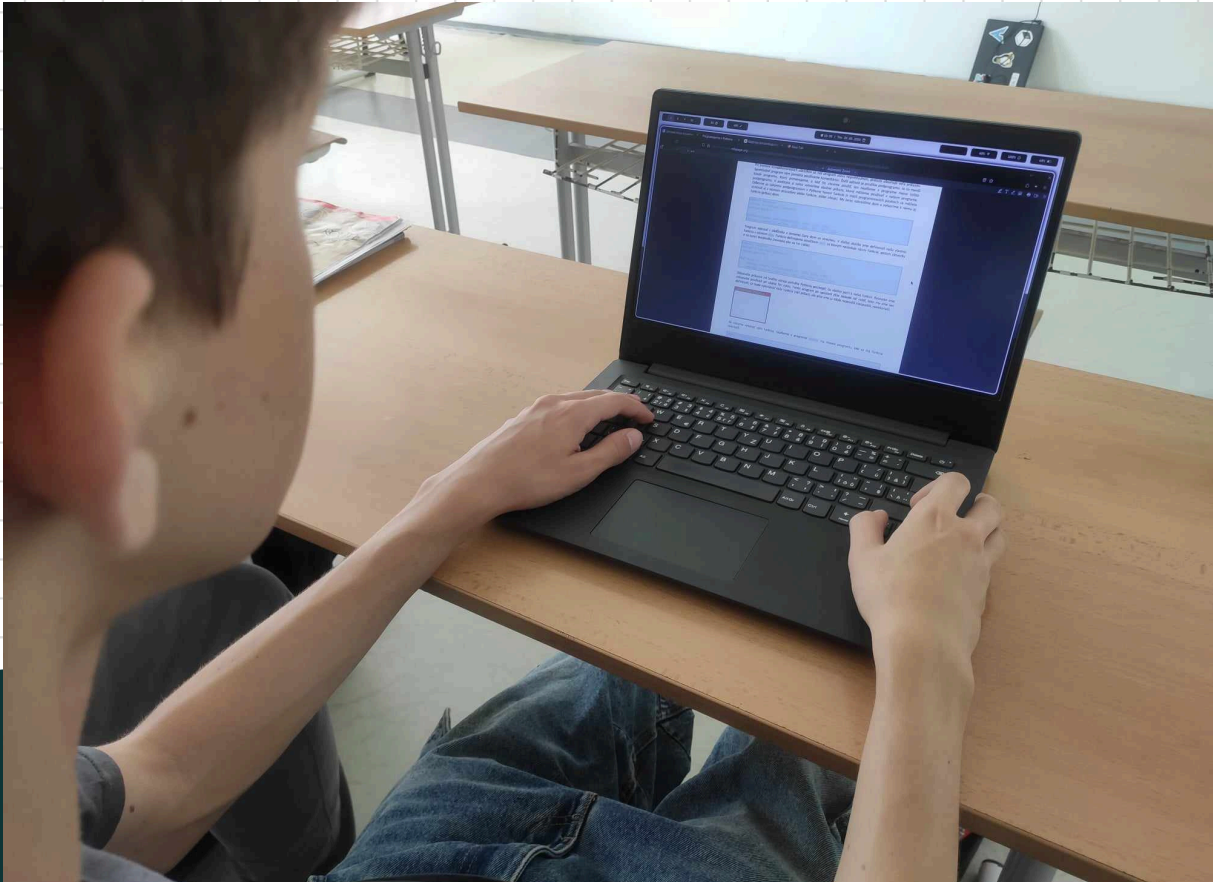




PRILOGA

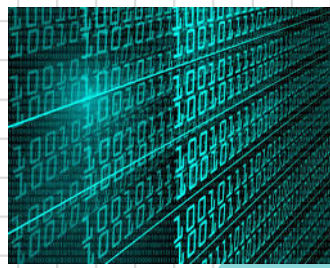


FOTOGRAFIJE IZ IZVEDBE LEKCIJE





PRILOGA



ŠTUDENTSKI REZULTAT

IZ UVOZA TKINTER *

USTVARITE OKNO TKINTER ROOT = TK()

USTVARITE PLATNO ZA RISANJE NA

PLATNU = CANVAS(ROOT, WIDTH=300,

HEIGHT=300) CANVAS.PACK() USTVARITE

KVADRAT

**CANVAS.CREATE_RECTANGLE(50, 50, 80 ,
80, FILL="BLUE")**

USTVARITE PRAVOKOTNIK

**CANVAS.CREATE_RECTANGLE(100, 100, 160,
140, FILL="RED")**

ZAŽENITE GLAVNO ZANKO

TKINTER ROOT.MAINLOOP()



UČNA PRIPRAVA



ZADEVA: PROGRAMIRANJE

TEMA LEKCIJE: PROGRAMIRANJE V PYTHONU – RISANJE IN BARVE.

UČNI CILJI:

- **1. RAZUMETI OSNOVNA NAČELA RISANJA V PYTHONU Z UPORABO KNJIŽNICE TKINTER**
- **2. NAUČITI SE UPORABLJATI RAZLIČNE BARVE IN UČINKE PRI RISANJU V PYTHONU.**

ORODJA:

**RAČUNALNIK Z NAMEŠČENIM PYTHONOM
IN KNJIŽNICO TKINTER.
PYTHON VZORČNI UČBENIŠKI DELOVNI
LIST Z VAJAMI**

**HOOK/INTRO: UPORABA RAZLIČNIH BARV
PRI RISANJU:**

**DOLOČITEV BARVE Z UPORABO VREDNOSTI
RGB**

**UPORABA RAZLIČNIH BARV ZA POSAMEZNE
RISANE PREDMETE**

- **USTVARJANJE UČINKOV Z BARVNIMI
KOMBINACIJAMI**

**STRUKTURA LEKCIJE: UVOD (4 MINUTE) – USTVARJANJE PREPROSTE SLIKE Z RISBO V
PYTHONU Z UPORABO BARV IN UČINKOV**

**POGOVOR Z UČENCI O OSNOVNIH PRINCIPIH RISANJA IN DELA Z BARVAMI V PYTHONU S
POMOČJO KNJIŽNICE TKINTER**

- **RAZISKOVANJE (12 MINUT) – RAZISKOVANJE UKAZOV NOVA SLIKA ZA POLNLENJE IN
ORIS**
- **PREDSTAVITEV (15 MINUT) – POVZETEK IN RAZLAGA ZNANJA IZ SKLOPA RAZISKOVANJE
PROBLEMATIKE IZDELAVE PROGRAMOV IZ OSNOVNIH PRINCIPOV RISANJA IN DELA Z
BARVAMI V PYTHON ELABORATION (9 MINUT) – PODPORA UČENCEM PRI NADALJNJEM
SAMOSTOJNEM RISANJU IN ODKRIVANJU MOŽNOSTI PROGRAMIRANJE V PYTHONU Z
UPORABO BARV**
- **OCENJEVANJE (5 MINUT) – REŠEVANJE TESTA ZA SAMOOCENJEVANJE, POGOVOR O
ODGOVORIH.**

DELOVNI LIST ZA UČENCA

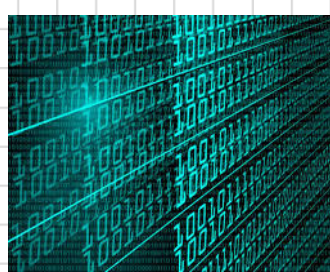
**1. NADALJEVALI BOMO PO NAŠEM ELEKTRONSKEM UČBENIKU PROGRAMIRANJE V
PYTHONU, KI JE NAMENJEN SREDNJIM ŠOLAM. VZOREC UČBENIKA NAJDETE NA:
[HTTP://WWW.PROGRAMUJEMVPTHONE.SK/UKAZKA.PDF](http://www.programujemvpythone.sk/ukazka.pdf) VZOREC VSEBUJE
ENOSTAVNE PRIMERE RISANJA V BARVAH:**

1. POLNLENJE OBLIKE:

**KROG, PRAVOKOTNIK, KVADRAT IN TRIKOTNIK ORIS RAZLIČNIH BARV IN DEBELIN ČRT
PREDOGLED KODE ZA POLNILO IN ORIS FAZA OSVETLITVE – OSVETLITEV NOVIH UKAZOV
– NA TABLI BOM NA KRATKO ZABELEŽIL OSNOVNE POJME**



UČNA PRIPRAVA

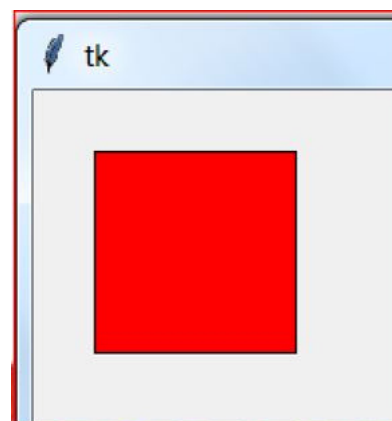


2. DELOVNI LIST UČENCEV

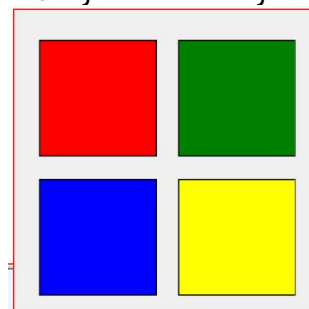
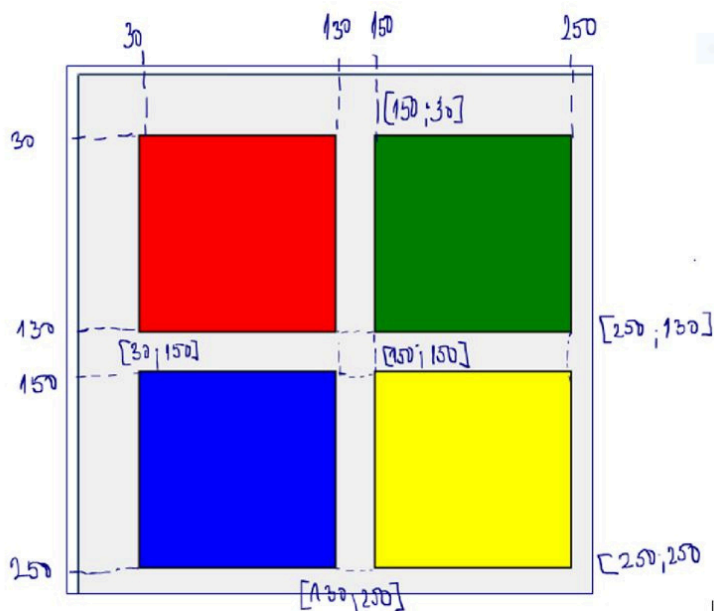
1. Naloga: Ustvarite nov program 06lek_p02.py, narišite barvni pravokotnik z naslednjo kodo:

```
*06lek_p02.py - F:/14_Python(SŠ)/PythonSŠ_PracovnéListy/06_PL (riešenia)/06lek_p02.py (3.6.2rc1)*  
File Edit Format Run Options Window Help  
import tkinter  
canvas = tkinter.Canvas()  
canvas.pack()  
  
canvas.create_rectangle(30, 30, 130, 130, fill='red')
```

Širina (in višina) kvadrata je 100

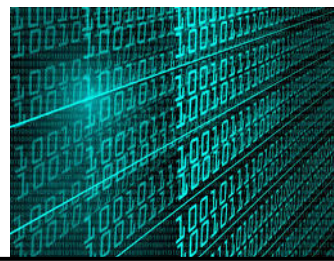


2. Naloga: V program 06lek_p03.py dodajte še 3 ukaze za risanje pravokotnikov, da dobite naslednjo sliko:
Širina (in višina) kvadrata je 100. Razdalja med kvadrati je 20





UČNA PRIPRAVA

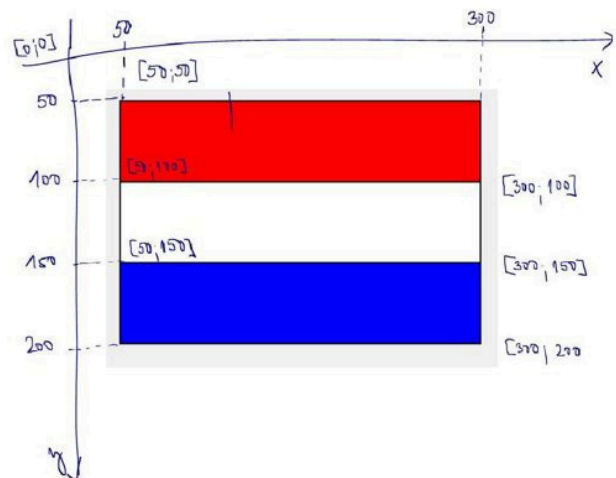
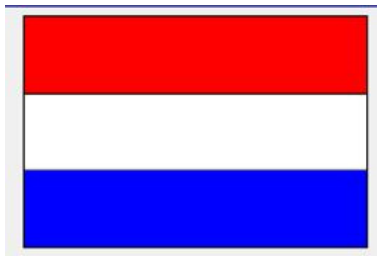


```
import tkinter
canvas = tkinter.Canvas()
canvas.pack()

canvas.create_rectangle(30, 30, 130, 130, fill='red')
canvas.create_rectangle(150, 30, 250, 130, fill='green')
canvas.create_rectangle(30, 150, 130, 250, fill='blue')
canvas.create_rectangle(150, 150, 250, 250, fill='yellow')
```

1.

3. Naloga: Ustvarite nov program 06lek_p04.py, ki nariše nizozemsko zastavo:



```
import tkinter
canvas = tkinter.Canvas()
canvas.pack()
```

```
canvas.create_rectangle(50, 50, 300, 100, fill='red')
canvas.create_rectangle(50, 100, 300, 150, fill='white')
canvas.create_rectangle(50, 150, 300, 200, fill='blue')
```

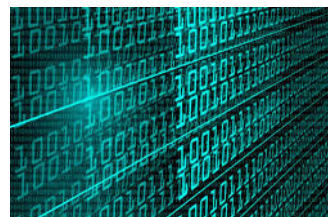
1.

4. Naloga: Ustvarite program 06lek_p05.py, ki nariše irsko zastavo z 'oranžno' barvo:



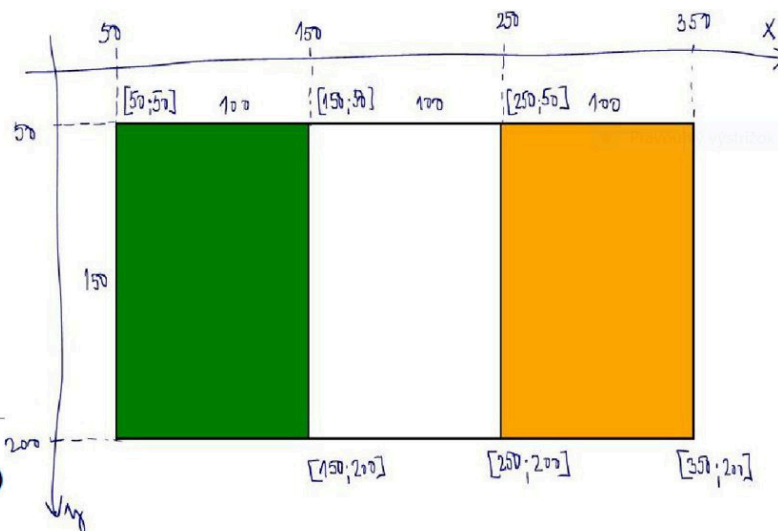


UČNA PRIPRAVA



1.

DELOVNI LIST ZA UČENCA

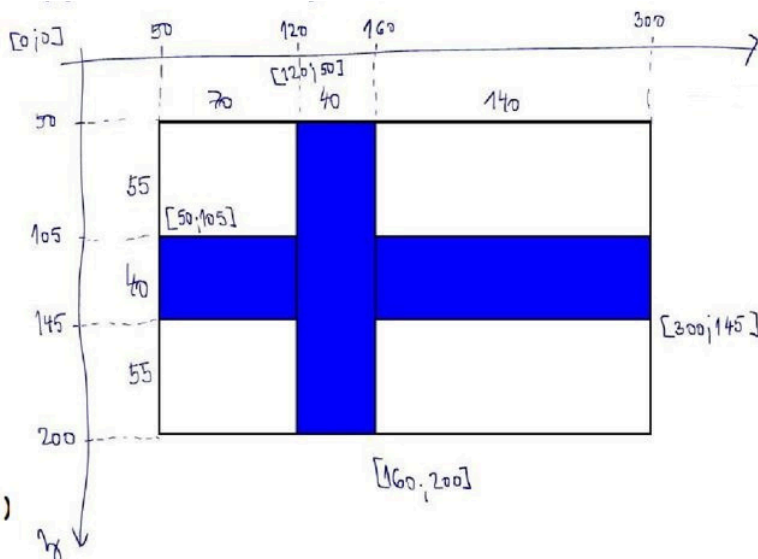


```
import tkinter
canvas = tkinter.Canvas()
canvas.pack()
```

```
canvas.create_rectangle(50, 50, 150, 200, fill='green')
canvas.create_rectangle(150, 50, 250, 200, fill='white')
canvas.create_rectangle(250, 50, 350, 200, fill='orange')
```

1.

5. Naloga: Ustvari nov program Oblek_p06.py in nariši sliko, kot je tale: Zastavi katere države je podobna slika?

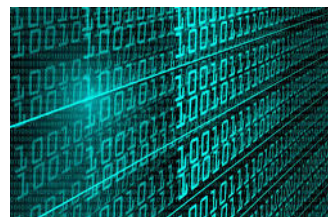


```
import tkinter
canvas = tkinter.Canvas()
canvas.pack()
```

```
canvas.create_rectangle(50, 50, 300, 200, fill='white')
canvas.create_rectangle(50, 105, 300, 145, fill='blue')
canvas.create_rectangle(120, 50, 160, 200, fill='blue')
```

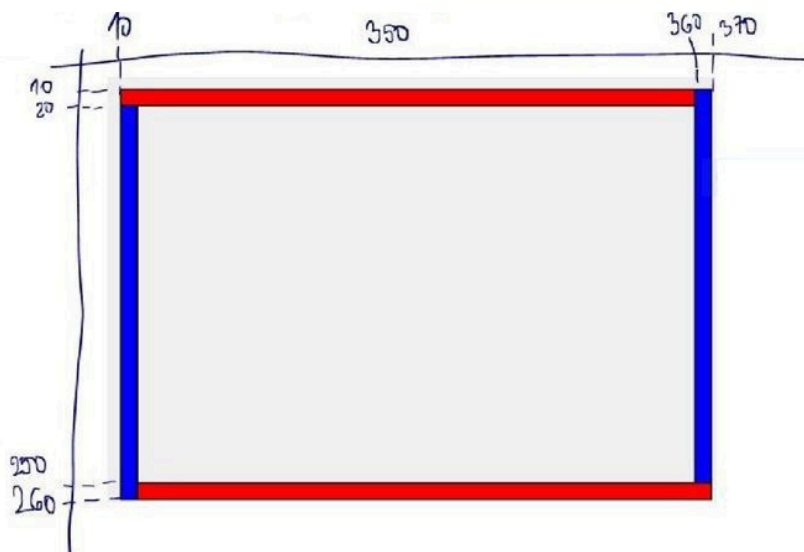


UČNA PRIPRAVA



1. DELOVNI LIST ZA UČENCA

1. 6. Naloga: • Ustvarite nov program 06lek_p07.py, v katerem iz štirih ozkih pravokotnikov narišete tak okvir:



```
import tkinter
canvas = tkinter.Canvas()
canvas.pack()

canvas.create_rectangle(10, 10, 360, 20, fill='red')
canvas.create_rectangle(10, 20, 20, 260, fill='blue')
canvas.create_rectangle(20, 250, 370, 260, fill='red')
canvas.create_rectangle(360, 10, 370, 250, fill='blue')
```

1. PRI RISANJU V PYTHONU IMAMO MOŽNOST UPORABE VELIKIH BARV. TUKAJ JE IZBOR NEKATERIH OD NJIH:

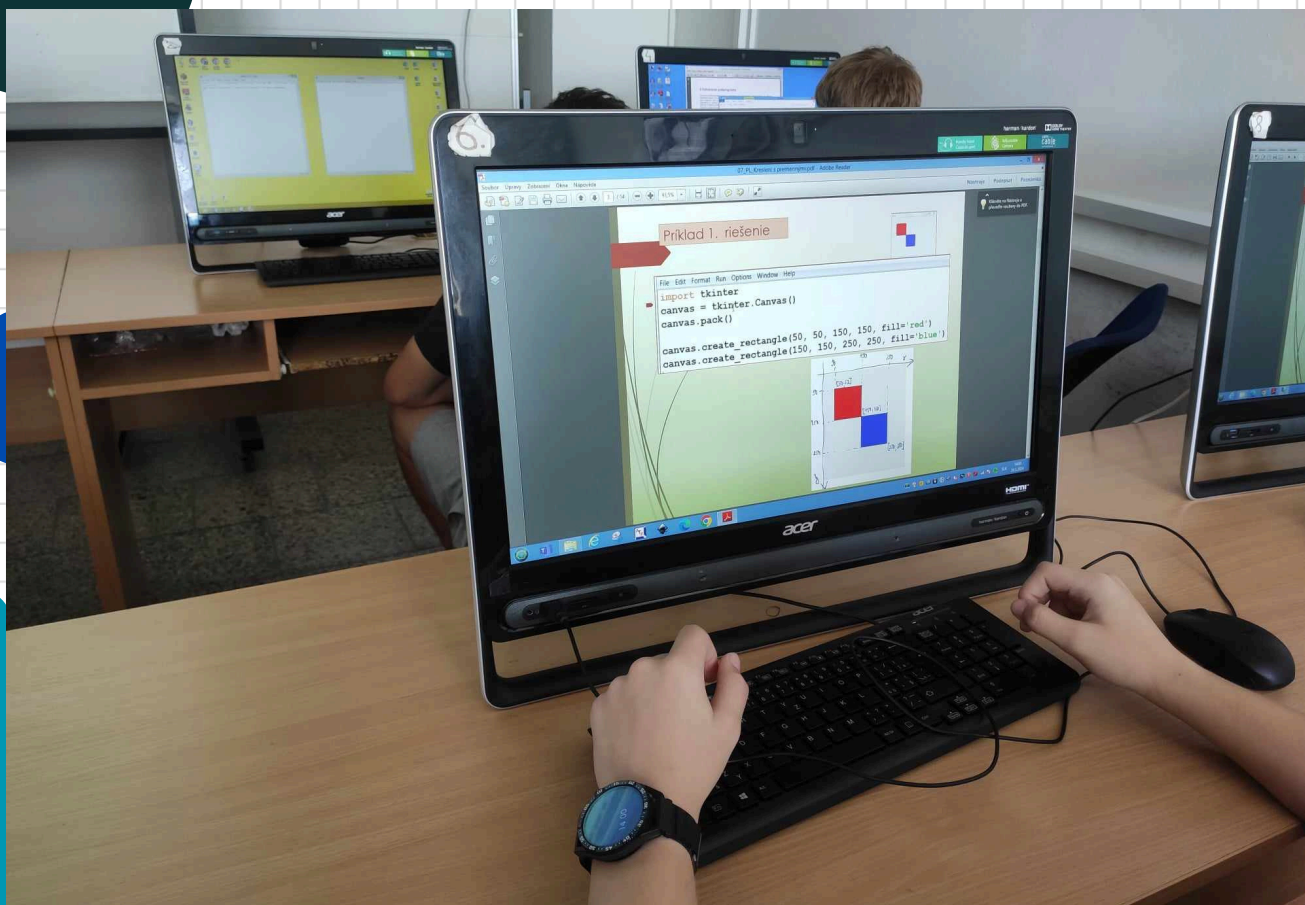
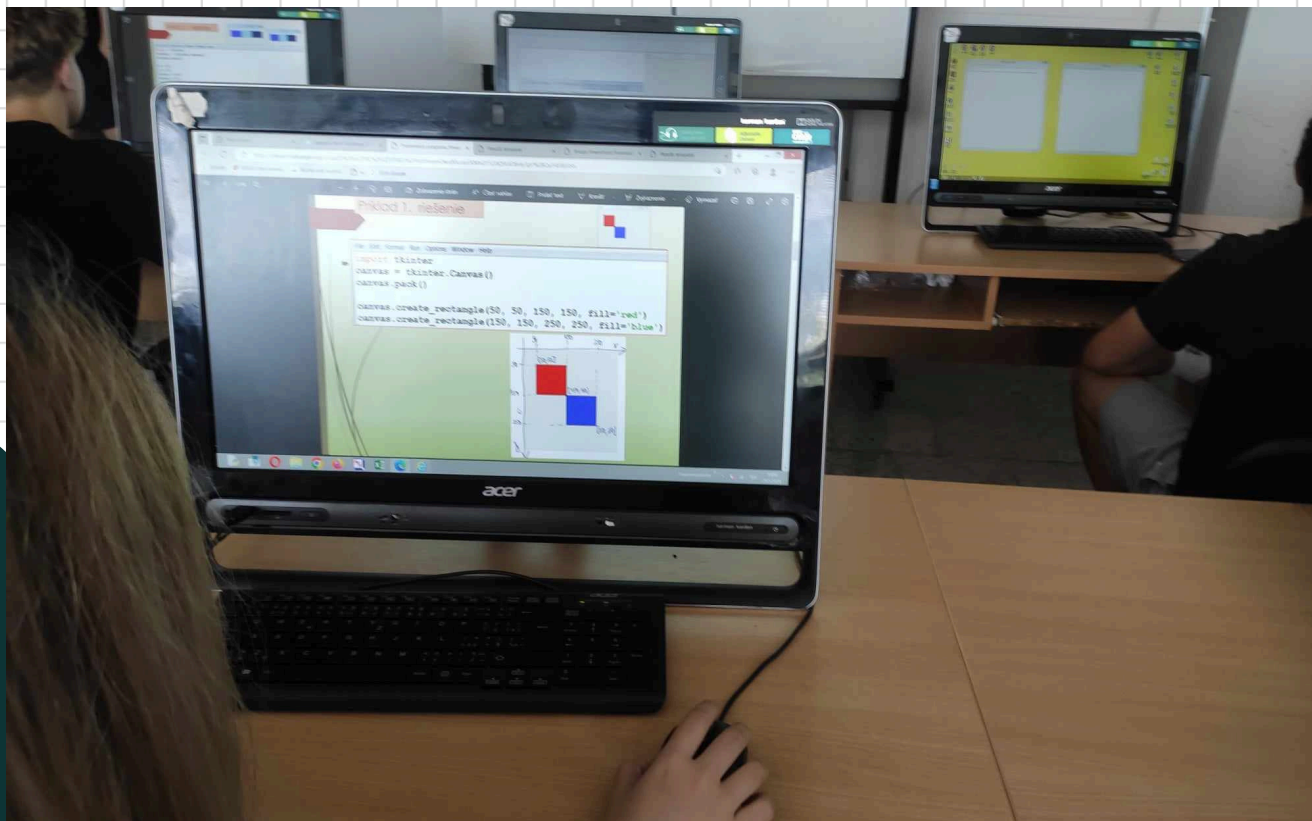
Blue	LightBlue	Cyan	SkyBlue	CornFlowerBlue	DeepSkyBlue	DodgerBlue
RoyalBlue	SlateBlue	SteelBlue	MediumBlue	Navy	Red	SandyBrown
Salmon	Coral	Tomato	Orange	DarkOrange	OrangeRed	IndianRed
Chocolate	Tan	Maroon	Sienna	Brown	SaddleBrown	Pink
Plum	Violet	Orchid	Magenta	Purple	DarkMagenta	Green
PaleGreen	YellowGreen	MediumSeaGreen	LawnGreen	LimeGreen	ForestGreen	DarkGreen
Yellow	Khaki	Gold	Gray	LightGray	Black	White



PRILOGA

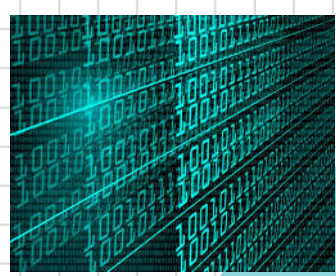


FOTOGRAFIJE IZ IZVEDBE LEKCIJE





PRILOGA



FOTOGRAFIJE IZ IZVEDBE LEKCIJE

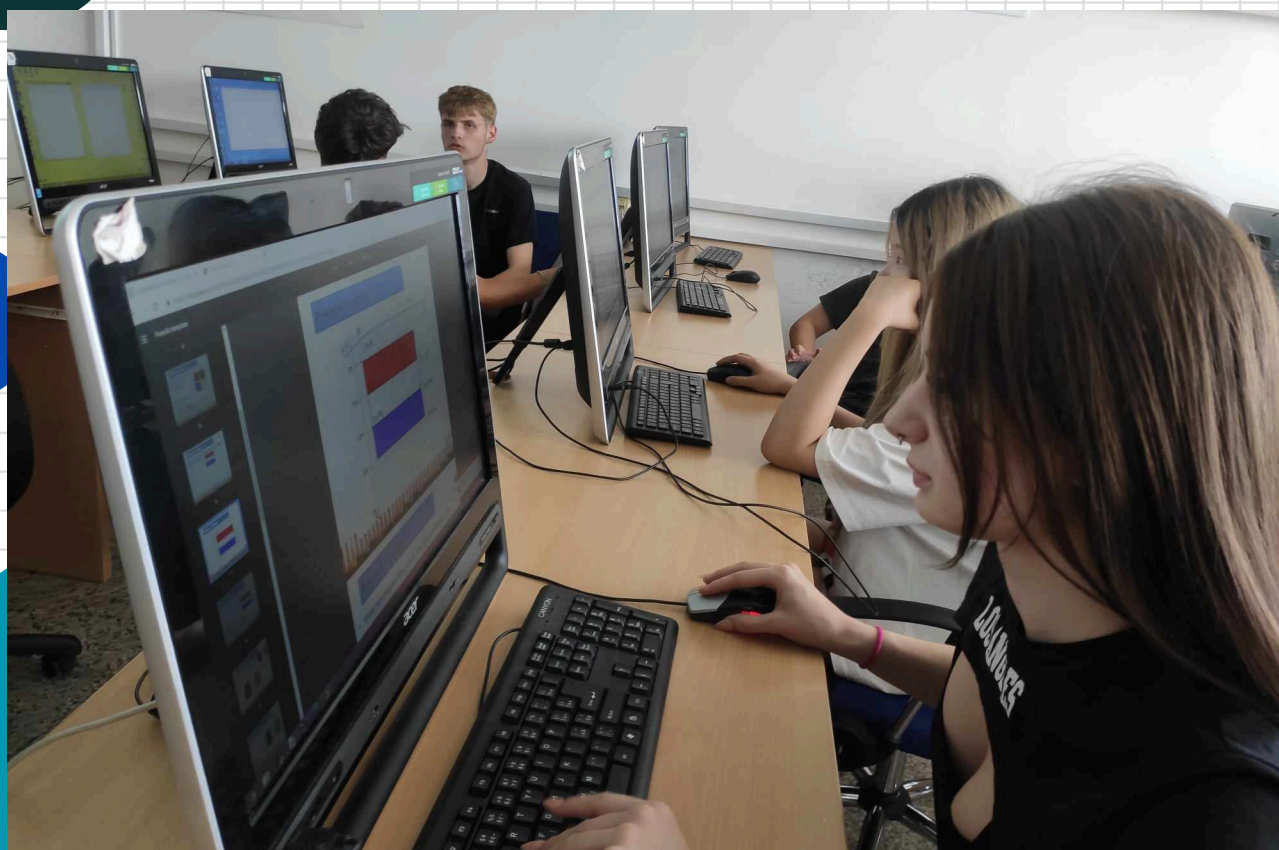
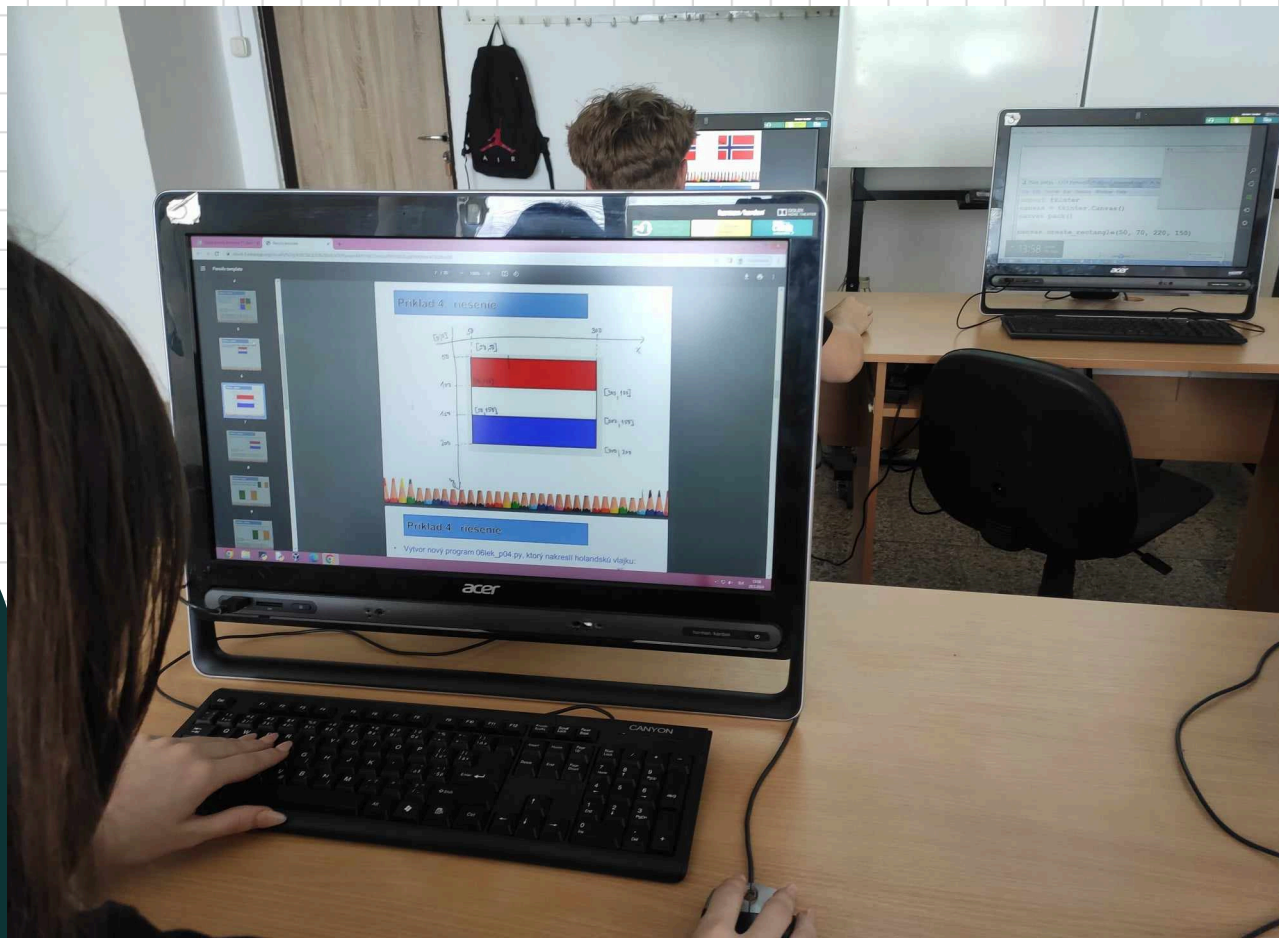




PRILOGA

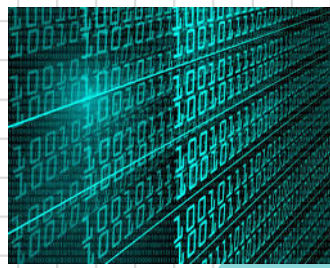


FOTOGRAFIJE IZ IZVEDBE LEKCIJE





PRILOGA



ŠTUDENTSKI REZULTAT

UVOZI TKINTER

```
# USTVARILI BOMO OKNO WINDOW = TKINTER.TK()  
WINDOW.TITLE("TRIJE PRAVOKOTNIKI")
```

```
# NASTAVIMO VELIKOST OKNA WIDTH = 400 HEIGHT  
= 200 WINDOW.GEOMETRY(F"{WIDTH}X{HEIGHT}")
```

```
# USTVARILI BOMO PLATNO CANVAS =  
TKINTER.CANVAS(WINDOW, WIDTH=WIDTH,  
HEIGHT=HEIGHT) CANVAS.PACK()
```

```
# PRVI PRAVOKOTNIK X1 = 50 Y1 = 20 X2 = 150 Y2 =  
80 COLOR1 = "RDEČE"  
CANVAS.CREATE_RECTANGLE(X1, Y1, X2, Y2,  
FILL=COLOR1)
```

```
# DRUGI PRAVOKOTNIK X2 = 200 Y2 = 50 COLOR2 =  
"MODRO" PLATNO.CREATE_RECTANGLE(X1, Y1, X2, Y2,  
FILL=COLOR2)
```

```
# TRETJI PRAVOKOTNIK X1 = 100 Y1 = 100 X2 = 250  
Y2 = 150 COLOR3 = "ZELENO"  
PLATNO.CREATE_RECTANGLE(X1, Y1, X2, Y2,  
FILL=COLOR3)
```




UČNA PRIPRAVA



ZADEVA: PROGRAMIRANJE

TEMA LEKCIJE: PROGRAMIRANJE V PYTHONU – RISANJE S SPREMENLJIVKAMI.

UČNI CILJI:

- **1. RAZUMETI OSNOVNA NAČELA RISANJA V PYTHONU Z UPORABO SPREMENLJIVK.**
- **2. NAUČITI SE UPORABLJATI SPREMENLJIVKE PRI RISANJU V PYTHONU.**

ORODJA:

**RAČUNALNIK Z NAMEŠČENIM PYTHONOM
IN KNJIŽNICO TKINTER.
PYTHON VZORČNI UČBENIŠKI DELOVNI
LIST Z VAJAMI**

HOOK/INTRO: UPORABA SPREMENLJIVK PRI RISANJU:

**DEFINIRANJE IN UPORABA SPREMENLJIVK V
PYTHONU
UPORABITE SPREMENLJIVKE ZA RISANJE
OBLIK.
DOLOČITE BARVO IN VELIKOST OBLIK.**

POTEK: UDEJSTVOVANJE (4 MINUTE) – USTVARJANJE SLIKE Z RISBO

- **V PYTHONU Z UPORABO SPREMENLJIVKE POGOVOR Z UČENCI O OSNOVNIH PRINCIPIH RISANJA IN DELA Z BARVAMI V PYTHONU S POMOČJO KNJIŽNICE TKINTER**
- **RAZISKOVANJE (12 MINUT) – RAZISKOVANJE NOVIH SPREMENLJIVK UKAZOV ZA POLNILO SLIKE IN ORIS**
- **INTERPRETACIJA (15 MINUT) – POVZEMANJE IN PISANJE ENOSTAVNEGA PROGRAMA V PYTHONU, KI RIŠE RAZLIČNE OBLIKE Z RAZLIČNIMI BARVAMI IN VELIKOSTMI IZDELAVA (9 MINUT) – PODPORA UČENCEM PRI NADALJNJEM SAMOSTOJNEM RISANJU MOŽNOSTI PROGRAMIRANJA V PYTHONU Z UPORABO SPREMENLJIVK**
- **OCENJEVANJE (5 MINUT) – REŠEVANJE TESTA ZA SAMOOCENJEVANJE, POGOVOR O ODGOVORIH.**

DELOVNI LIST ZA UČENCA

**1. NADALJEVALI BOMO PO NAŠEM ELEKTRONSKEM UČBENIKU PROGRAMIRANJE V PYTHONU, KI JE NAMENJEN SREDNJIM ŠOLAM. VZOREC UČBENIKA NAJDETE NA:
[HTTP://WWW.PROGRAMUJEMEVPYTHONE.SK/UKAZKA.PDF](http://www.programujemevpythone.sk/ukazka.pdf) VZOREC VSEBUJE
PREPROSTE PRIMERE RISANJA S SPREMENLJIVKO:**

- 1. PROGRAM, KI NARIŠE KVADRAT, PRAVOKOTNIK, TRIKOTNIK IN KROG. KOORDINATE ZGORNJEGA LEVEGA KOTA PREDMETA SO SHRANJENE V SPREMENLJIVKAH X, Y.**
- 2. VZORČNA KODA S SPREMENLJIVKAMI ZA POLNILO IN OBRIS RAZLIČNIH BARV IN DEBELIN ČRT FAZA IZPOSTAVITVE – IZPOSTAVITEV NOVIH STAVKOV S SPREMENLJIVKAMI – NA TABLI SI BOM NA KRATKO ZABELEŽIL TERMIN ZEMLJEVID OSNOVNIH POJMOV**

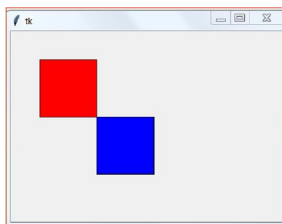


UČNA PRIPRAVA



3. DELOVNI LIST UČENCEV

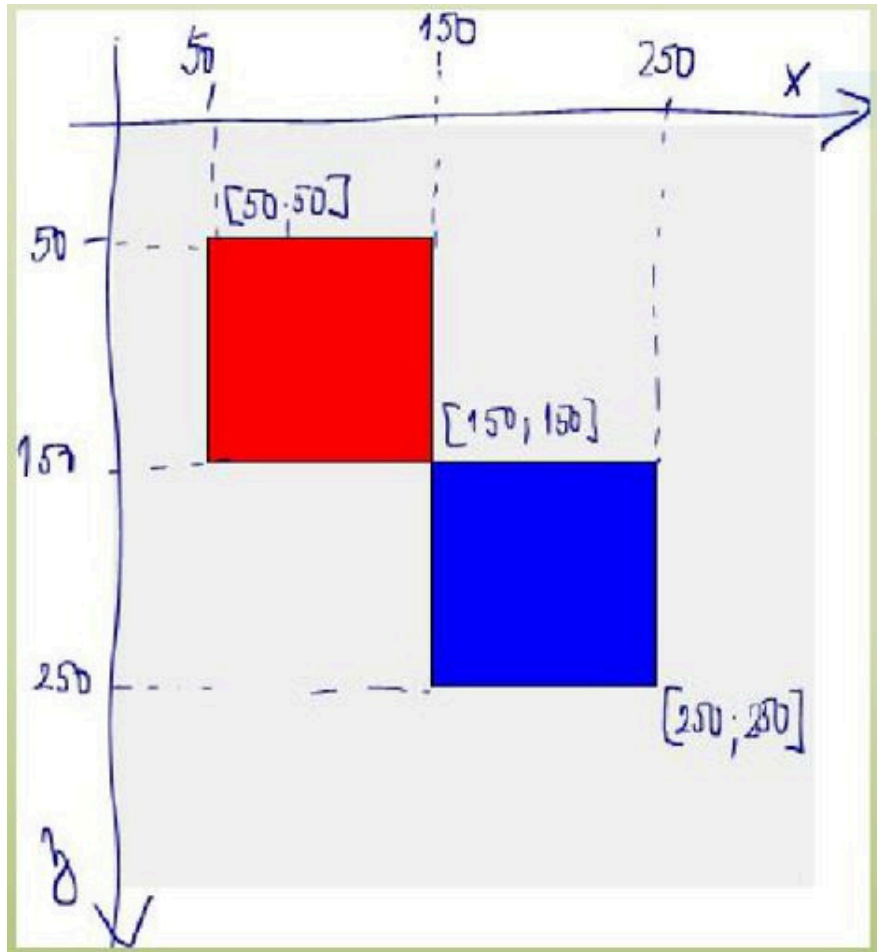
1. Naloga: Ustvari nov program 07lek_p01.py, ki bo narisal dva dotikajoča se kvadrata, kot je prikazano na sliki. Oba imata stransko dolžino 100, rdeča pa ima zgornji levi kot na konici [50, 50].



File Edit Format Run Options Window Help

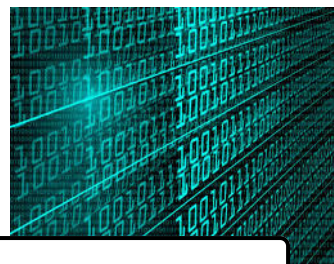
```
import tkinter
canvas = tkinter.Canvas()
canvas.pack()

canvas.create_rectangle(50, 50, 150, 150, fill='red')
canvas.create_rectangle(150, 150, 250, 250, fill='blue')
```





UČNA PRIPRAVA



3. DELOVNI LIST UČENCEV

1.

1.

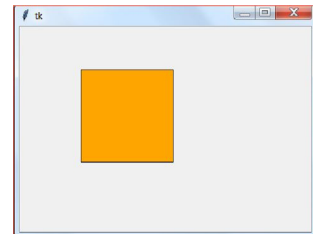
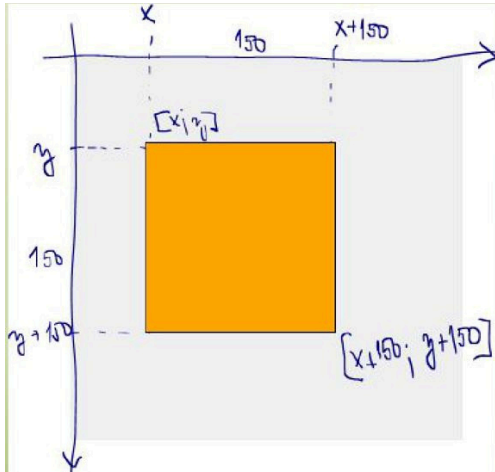
2. Naloga: Ustvarite nov program 07lek_p02.py, ki nariše kvadrat s stranico 150.

2.

Spremenljivki x , y shranita koordinate zgornjega levega kota kvadrata.

3.

Koordinate zgornjega levega kota $[x;y] = [100;70]$ so vnesene kot spremenljivke.



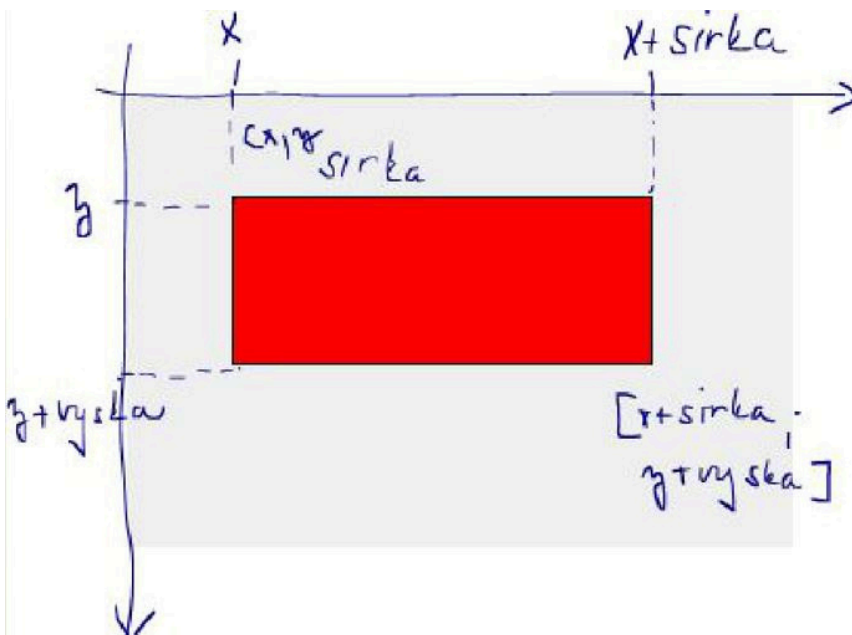
```
import tkinter
canvas = tkinter.Canvas()
canvas.pack()

x = 100
y = 70
canvas.create_rectangle(x,y,x + 150,y + 150, fill='orange')
```

1. 3. Naloga: Izdelajte program 07lek_p03.py, ki nariše pravokotnik z zgornjim levim kotom na x , y koordinatah glede na širino in višino. Izberite barvo po želji.

2.

Na primer, ko program vsebuje: $x = 100$, $y = 70$, širina = 200, višina = 80, se nariše pravokotnik kot na sliki:





UČNA PRIPRAVA



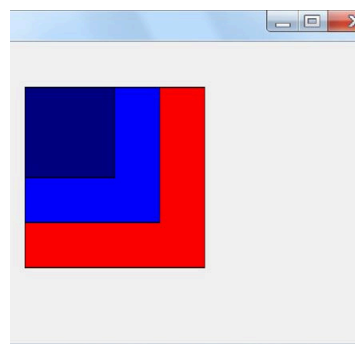
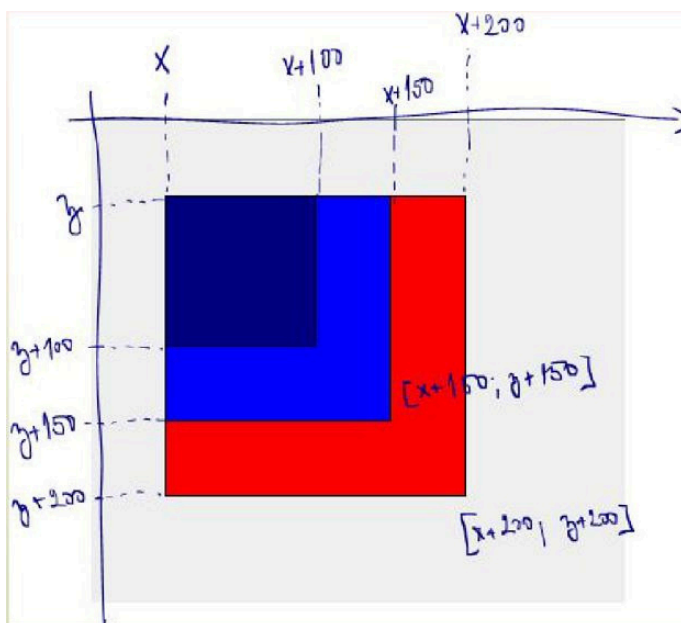
1. 3. DELOVNI LIST UČENCEV

File Edit Format Run Options Window Help

```
import tkinter
canvas = tkinter.Canvas()
canvas.pack()

x = 100
y = 70
sirka = 200
vyska = 80
canvas.create_rectangle(x, y, x + sirka, y + vyska, fill='red')
```

1. 4. Naloga: Ustvarite program 07lek_p04.py, ki riše kvadrate, kot je prikazano na sliki. Ti kvadrati imajo skupen zgornji levi kot, katerega koordinate so v spremenljivkah x, y.
2. Kvadrati se postopoma manjšajo, tako da ima rdeči dolžino stranice 200, modri 150 in temno moder 100.
3. Koordinate središča so lahko na primer: [x;y] = [80;50]



File Edit Format Run Options Window Help

```
import tkinter
canvas = tkinter.Canvas()
canvas.pack()

x = 80
y = 50
canvas.create_rectangle(x, y, x+200, y+200, fill='red')
canvas.create_rectangle(x, y, x+150, y+150, fill='blue')
canvas.create_rectangle(x, y, x+100, y+100, fill='navy')
```



UČNA PRIPRAVA

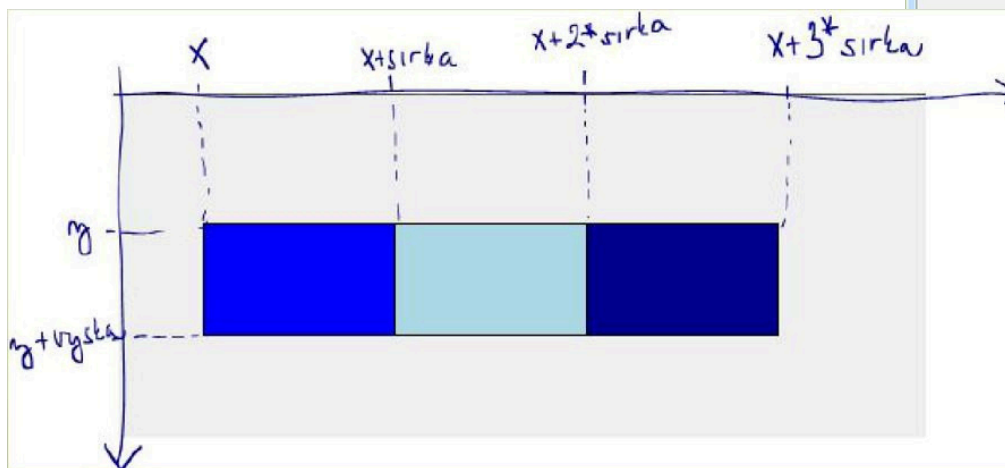
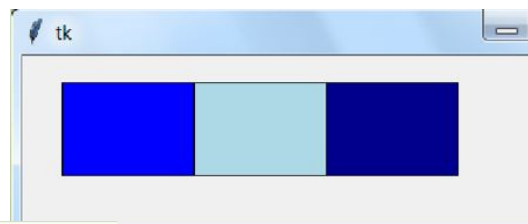


1. 3. DELOVNI LIST UČENCEV

1. 5. Naloga: Ustvarite program 07lek_p05.py, ki nariše tri pravokotnike, ki se dotikajo drug drugega:
2. Koordinate zgornjega levega kota prvega pravokotnika so shranjene v spremenljivkah x , y .

1. Vsi trije pravokotniki imajo enako širino in višino – te dimenzije so shranjene v spremenljivkah širina in višina .

1. Ali bo program deloval pravilno, tudi če povečate vrednost spremenljivke width za 20 in vrednost spremenljivke y za 10?
Če ne, program za popravilo.



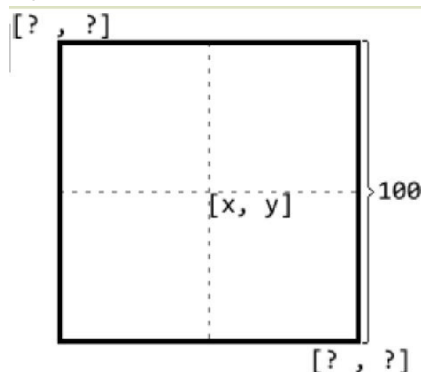
File Edit Format Run Options Window Help

```
import tkinter
canvas = tkinter.Canvas()
canvas.pack()

x = 30
y = 20
sirka = 100
vyska = 70
canvas.create_rectangle(x, y, x+sirka, y+vyska, fill='blue')
canvas.create_rectangle(x+sirka, y, x+2*sirka, y+vyska, fill='light blue')
canvas.create_rectangle(x+2*sirka, y, x+3*sirka, y+vyska, fill='dark blue')
```

Naloga: Ustvarite nov program 07-lek_p06 .py, ki nariše kvadrat, katerega središče ima koordinate x , y , stranice pa dolžine 100. Koordinate x , y so shranjene v spremenljivkah z istim imenom.

Če želite narisati ta kvadrat, morate izračunati koordinate njegovega zgornjega levega in spodnjega desnega kota. V novem programu napišite kodo, ki nariše zelen kvadrat s središčem $[x, y]$ in dolžino stranice 100.





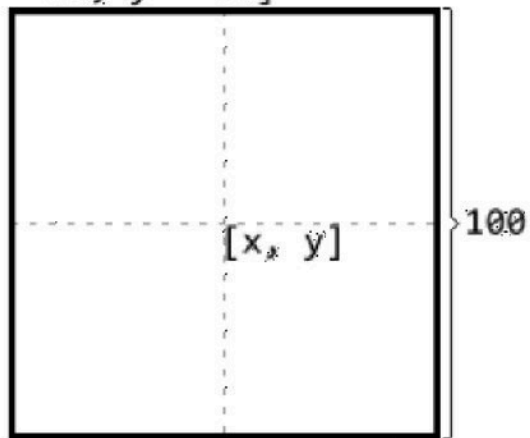
UČNA PRIPRAVA



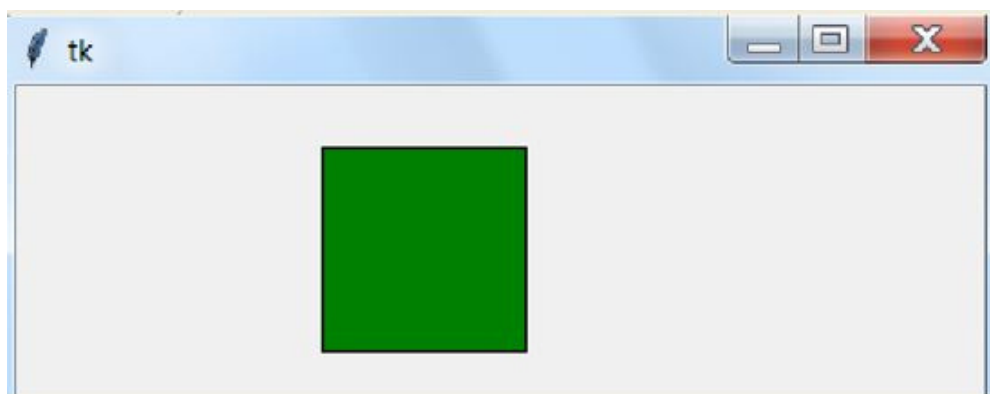
1.

3. DELOVNI LIST UČENCEV

$[x - 50, y - 50]$



$[x + 50, y + 50]$



File Edit Format Run Options Window Help

```
import tkinter
canvas = tkinter.Canvas()
canvas.pack()

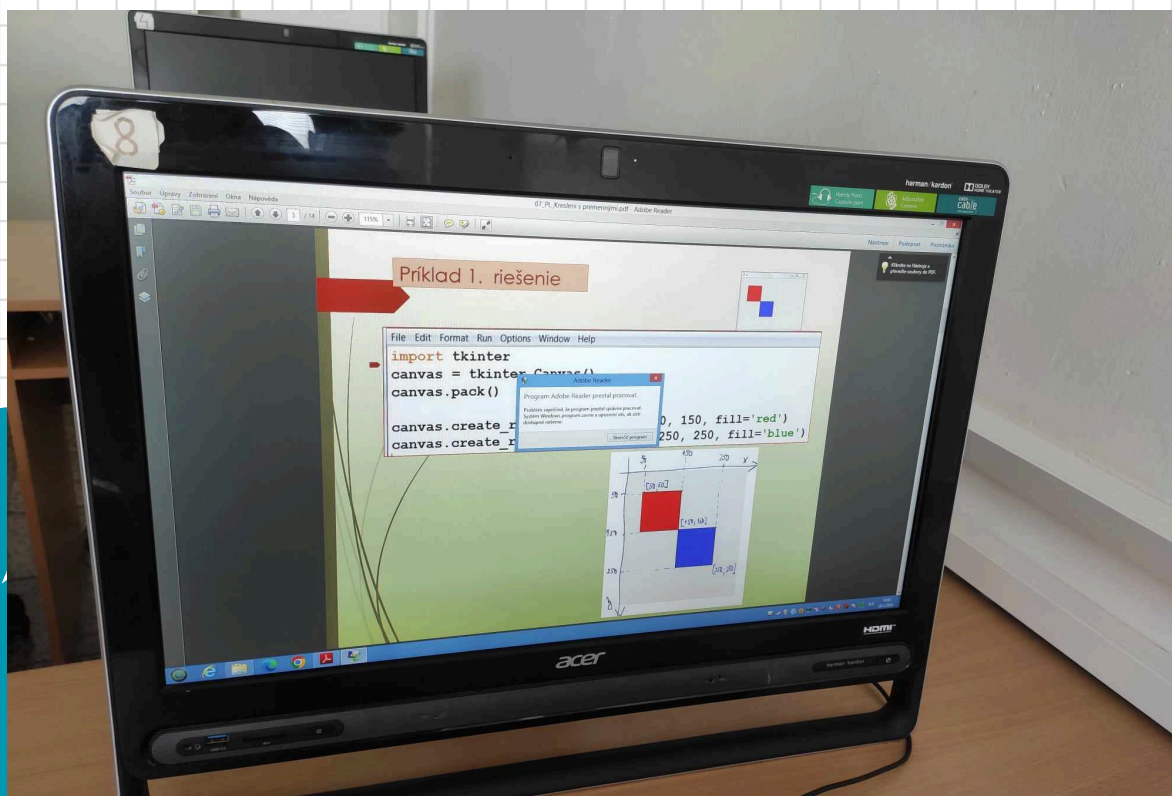
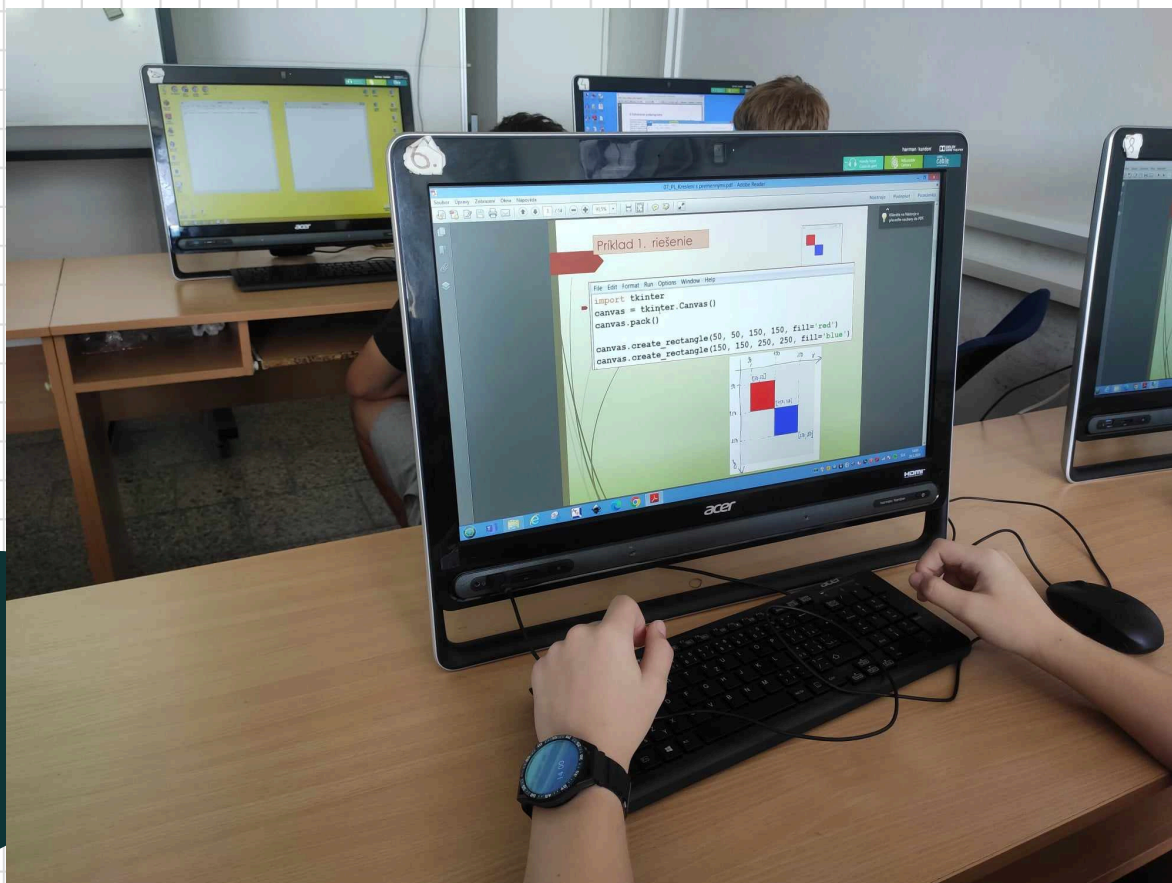
x = 200
y = 80
canvas.create_rectangle(x-50,y-50,x+50,y+50,fill='green')
```




PRILOGA

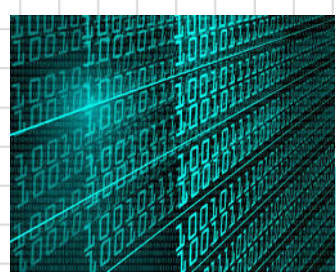


FOTOGRAFIJE IZ IZVEDBE LEKCIJE

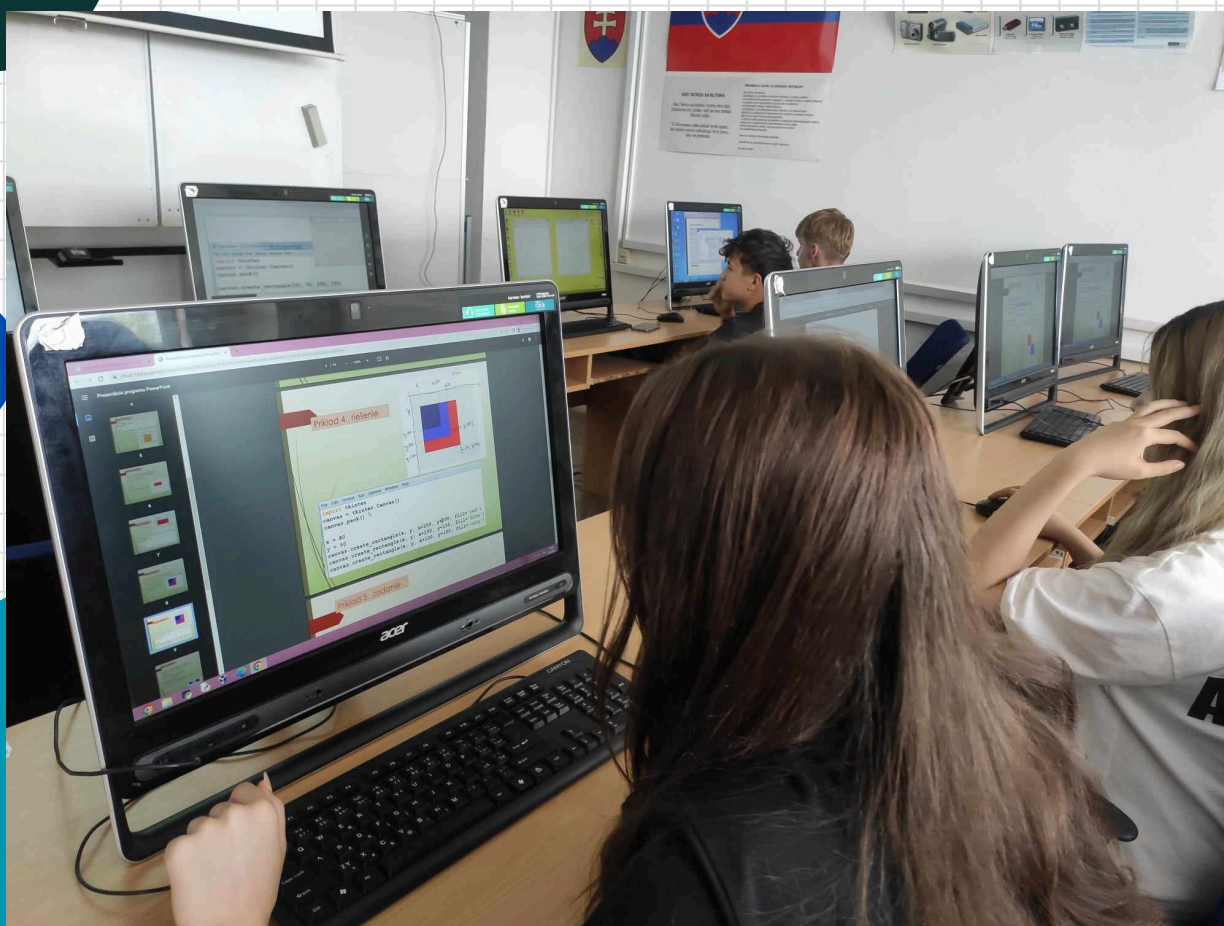
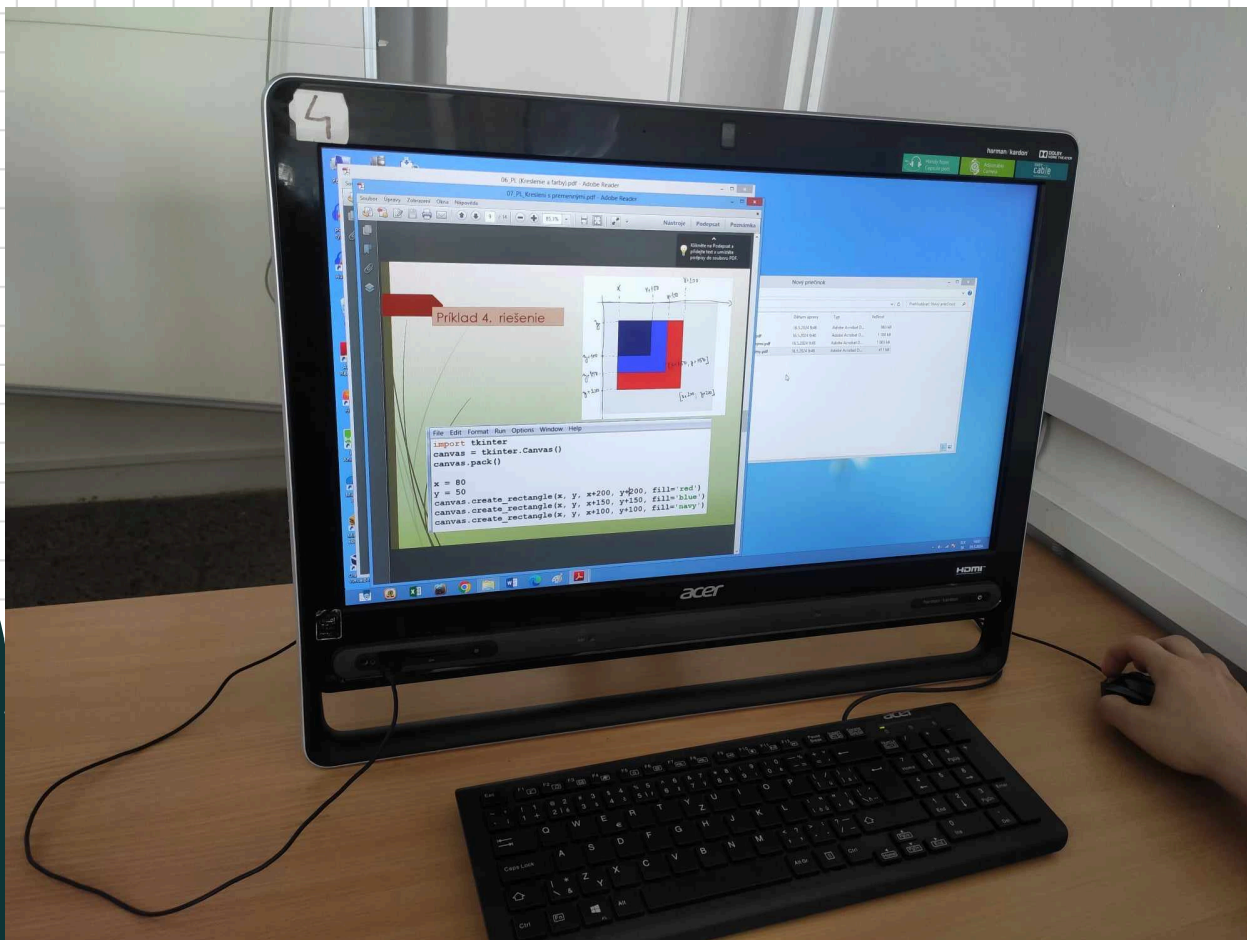




PRILOGA

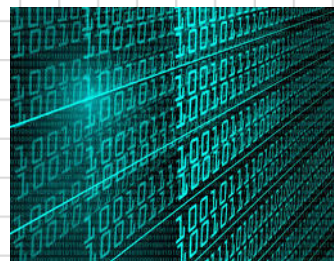


FOTOGRAFIJE IZ IZVEDBE LEKCIJE





PRILOGA



REZULTAT ŠTUDENTOV:

**NAPIŠITE PROGRAM, KI PREBERE TRI BESEDE IN NATISNE VSEH
6 RAZLIČNIH PERMUTACIJ. PO ZAGONU LAHKO DOBITE:**

VNESITE 1. BESEDO: BELA

VNESITE 2. BESEDO: MODRA

VNESITE 3. BESEDO: RDEČA

BELA MODRA RDEČA

BELA RDEČA MODRA

MODRA BELA RDEČA

MODRA RDEČA BELA

RDEČA BELA MODRA

RDEČA MODRA BELA

ALI UPORABITE SPREMENLJIVKE:

ZADAJ 1. SLOVO: X

ZADAJ 2. SLOVO: ***

ZADAJ 3. SLOVO: :-)

X * :-)**

X :-) ***

***** X :-)**

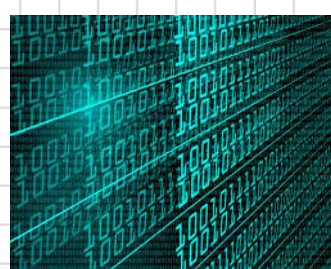
***** :-) X**

:-) X ***

:-) * X**



NAČRT LEKCIJE



ZADEVA: PROGRAMIRANJE

TEMA LEKCIJE: PROGRAMIRANJE V PYTHONU – PODPROGRAMI.

UČNI CILJI:

- **1. UPORABITE PODPROGRAME ZA POJASNITEV UKAZOV ZA RISANJE V PYTHONU**
- **2. NAUČITE SE DEFINIRATI FUNKCIJ MED RISANJEM V PYTHONU.**

ORODJA:
RAČUNALNIK Z NAMEŠČENIM PYTHONOM
IN KNJIŽNICO TKINTER.
PYTHON VZORČNI UČBENIŠKI DELOVNI
LIST Z VAJAMI

HOOK/INTRO: UPORABA RAZLIČNIH BARV PRI
RISANJU:

DEFINIRANJE FUNKCIJ

- **UPORABA PODPROGRAMA ZA POSAMEZNE**
RISANE PREDMETE
- **SPREMENITE FUNKCIJO, TAKO DA LAHKO**
DOLOČIMO LOKACIJO, KO JO UPORABLJAMO
- **PONOVITE UKAZE V ZANKI**

POTEK: POVEZAVA (4 MINUTE) – PODPROGRAM JE MANJŠI DEL PROGRAMA, KI GA
POIMENUJEMO, IN KO GA ŽELIMO UPORABITI, SAMO ZAPIŠEMO IME TEGA PODPROGRAMA V
PROGRAM.

POGOVOR Z UČENCI O PREDNOSTIH UPORABE PODPROGRAMOV V PYTHONU: – UPORABA
PODPROGRAMOV PRI RISANJU ZAHTEVNEJŠIH SLIK JE PRIMEREN, DA PROGRAM NE POSTANE
ZMEDEN, SAJ VSEBUJE VELIKO UKAZOV. UPORABA KOMENTARJEV NAM POMAGA NAREEDITI
PROGRAM BOLJ PREGLEDEN. BOLJŠI NAČIN JE UPORABA PODPROGRAMA.

• **RAZISKOVANJE (12 MINUT) – RAZISKOVANJE NOVIH UKAZOV – FUNKCIJA JE POIMENOVAN BLOK**
UKAZOV (VČASIH IMENOVAN PODPROGRAM). OPIŠEMO (DEFINIRAMO) GA S POSEBNO
KONSTRUKCIJO

• **PREDSTAVITEV (15 MINUT) – POVZETEK IN RAZLAGA ZNANJA IZ SKLOPA RAZISKOVANJE**
PROBLEMATIKE IZDELAVE PODPROGRAMOV V PYTHON DEVELOPMENT (9 MINUT) – PODPORA
ŠTUDENTOM PRI NADALJNJEM SAMOSTOJNEM USTVARJANJU PODPROGRAMOV V PYTHONU.

• **OCENJEVANJE (5 MINUT) – REŠEVANJE TESTA ZA SAMOOCENJEVANJE, POGOVOR O**
ODGOVORIH.

DELOVNI LIST ZA UČENCA

NADALJEVALI BOMO PO NAŠEM ELEKTRONSKEM UČBENIKU PROGRAMIRANJE V PYTHONU, KI JE
NAMENJEN SREDNJIM ŠOLAM. VZOREC UČBENIKA NAJDETE NA:

[HTTP://WWW.PROGRAMUJEMEVPYTHONE.SK/UKAZKA.PDF](http://www.programujemevpythone.sk/ukazka.pdf) VZOREC VSEBUJE PREPROSTE PRIMERE
USTVARJANJA PODPROGRAMOV V PYTHONU:

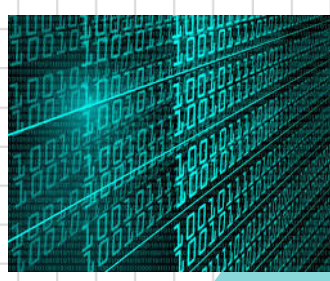
1. USTVARILI BOMO LASTNE UKAZE, KI JIH BOMO LAHKO UPORABILI V NAŠEM PROGRAMU.

TEHNIČNO SE TAKŠNI PODPROGRAMI V PYTHONU IMENUJEJO FUNKCIJE (V DRUGIH PROGRAMSKIH
JEZIKIH LAHKO SREČATE TUDI IME PROCEDURE ALI FUNKCIJE ALI OBOJE).

FAZA IZPOSTAVITVE – POTISKANJE STAVKOV STRAN OD LEVEGA ROBA POMAGA PYTHONU
RAZUMETI, KAJ VSE SPADA V NAŠO FUNKCIJO. – NA TABLI BOM NA KRATKO ZABELEŽIL OSNOVNE
IZRAZE



NAČRT LEKCIJE



4. DELOVNI LIST UČENCEV

1. Naloga: Nariši hišo in zanjo izdelaj funkcijsko (ukazno) hišo.

uvozi tkinter

```
canvas = tkinter.Canvas() canvas.pack() canvas.create_rectangle(100, 100, 150, 150) canvas.create_line(100, 100, 125, 50, 150, 100)
```

Program iz pravokotnika in lomljene črte nariše hišo s streho. V naslednjem primeru smo definirali lastno funkcijo, imenovano hiša.

Funkcijo definiramo z besedo `def`, ki ji sledi ime funkcije, nato oklepaj in na koncu dvopičje (tako kot pri zanki `for`).

```
uvozi tkinter canvas = tkinter.Canvas() canvas.pack() def dom():
```

```
canvas.create_rectangle(100, 100, 150, 150) canvas.create_line(100, 100, 125, 50, 150, 100)
```

Če želimo to funkcijo izvršiti, vpišemo v program `dom()` (na mesto programa, kjer naj se funkcija izvaja).

```
uvozi tkinter canvas = tkinter.Canvas() canvas.pack() def dom():
```

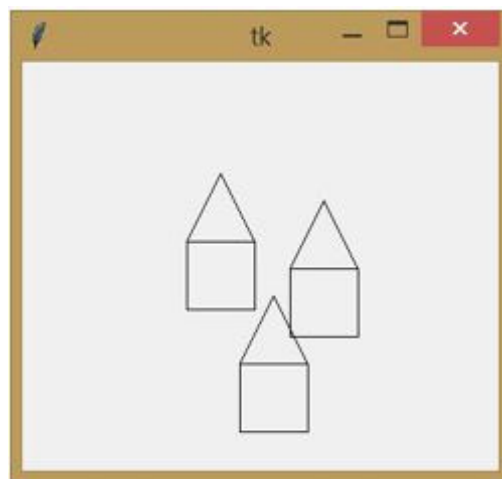
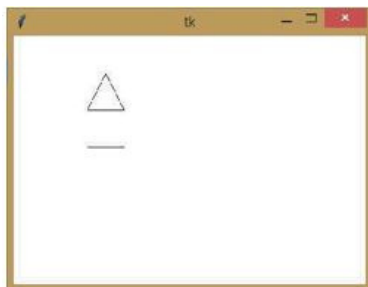
```
canvas.create_rectangle(100, 100, 150, 150) canvas.create_line(100, 100, 125, 50, 150, 100) dom()
```

Ta program zažene tudi funkcijo `dom()`. Pomen definiranja ukaza po meri lahko bolje vidimo, če spremenimo funkcijo hiše, da nariše hišo na naključni lokaciji.

```
uvozi tkinter iz naključnega uvoza * canvas = tkinter.Canvas() canvas.pack() def dom():
```

```
x = randint(100, 300) y = randint(100, 300) canvas.create_rectangle(x, y, x+50, y+50) canvas.create_line(x, y, x+25, y-50, x+50, y) dom() dom() dom()
```

V tem programu bomo že trikrat izvedli funkcijo `house()` in tako bodo na naključnem mestu izrisane tri hiše.





NAČRT LEKCIJE



4. DELOVNI LIST UČENCEV

2. Naloga: Uporabite zanko for, da boste zlahka narisali 10 hiš.

uvozi tkinter

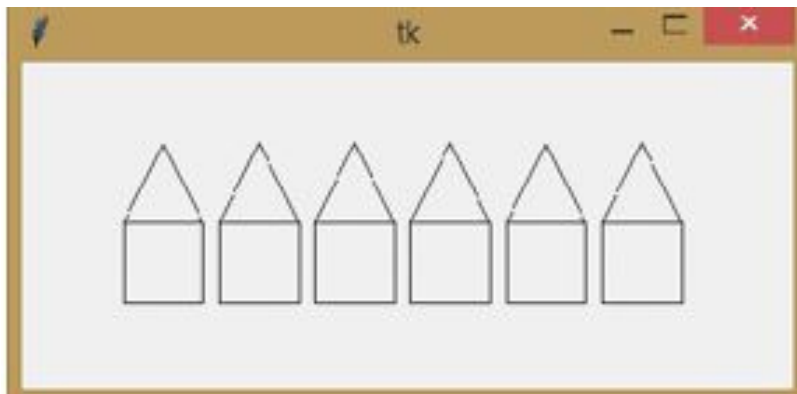
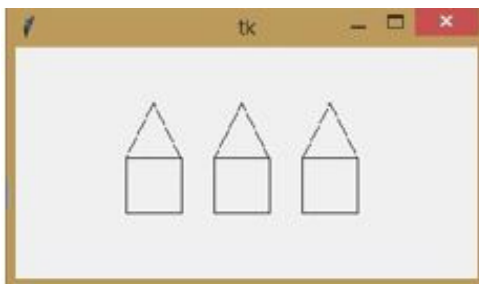
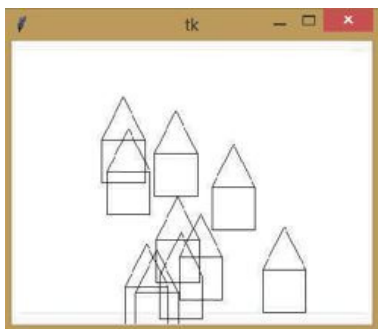
```
iz naključnega uvoza * canvas = tkinter.Canvas() canvas.pack() def dom():  
x = randint(100, 300) y = randint(100, 300) canvas.create_rectangle(x, y, x+50, y+50) canvas.create_line(x, y, x+25,  
y-50, x+50, y) za i v obsegu (1, 11):  
dom()
```

Funkcijo lahko tudi spremenimo tako, da ob uporabi vnesemo mesto, kjer želimo narisati hišo. Podobno kot pri uporabi funkcije `create_rectangle` številke v oklepajih vplivajo na lokacijo risanja. Takšna števila (ali drugi podatki), ki vplivajo na funkcijo, se imenujejo parametri. Pri definiranju funkcije s parametrom damo imena posameznih parametrov v oklepaj, ločena z vejico (če jih želimo imeti več). Ta imena lahko uporabimo v naši funkciji (v bistvu so nekakšne spremenljivke, ki jih lahko uporabimo le v tej funkciji) in imajo vrednost, ki smo jo vnesli pri uporabi funkcije.

uvozi tkinter

```
iz naključnega uvoza * canvas = tkinter.Canvas() canvas.pack() def dom(x, y):  
canvas.create_rectangle(x, y, x+50, y+50) canvas.create_line(x, y, x+25, y-50, x+50, y) dom(100, 100) dom(180, 100) dom(260,  
100)
```

```
uvozi tkinter iz naključnega uvoza * canvas = tkinter.Canvas() canvas.pack() def dom(x, y):  
canvas.create_rectangle(x, y, x+50, y+50) canvas.create_line(x, y, x+25, y-50, x+50, y) za i v obsegu (1, 7):  
dom(i*60, 100)
```

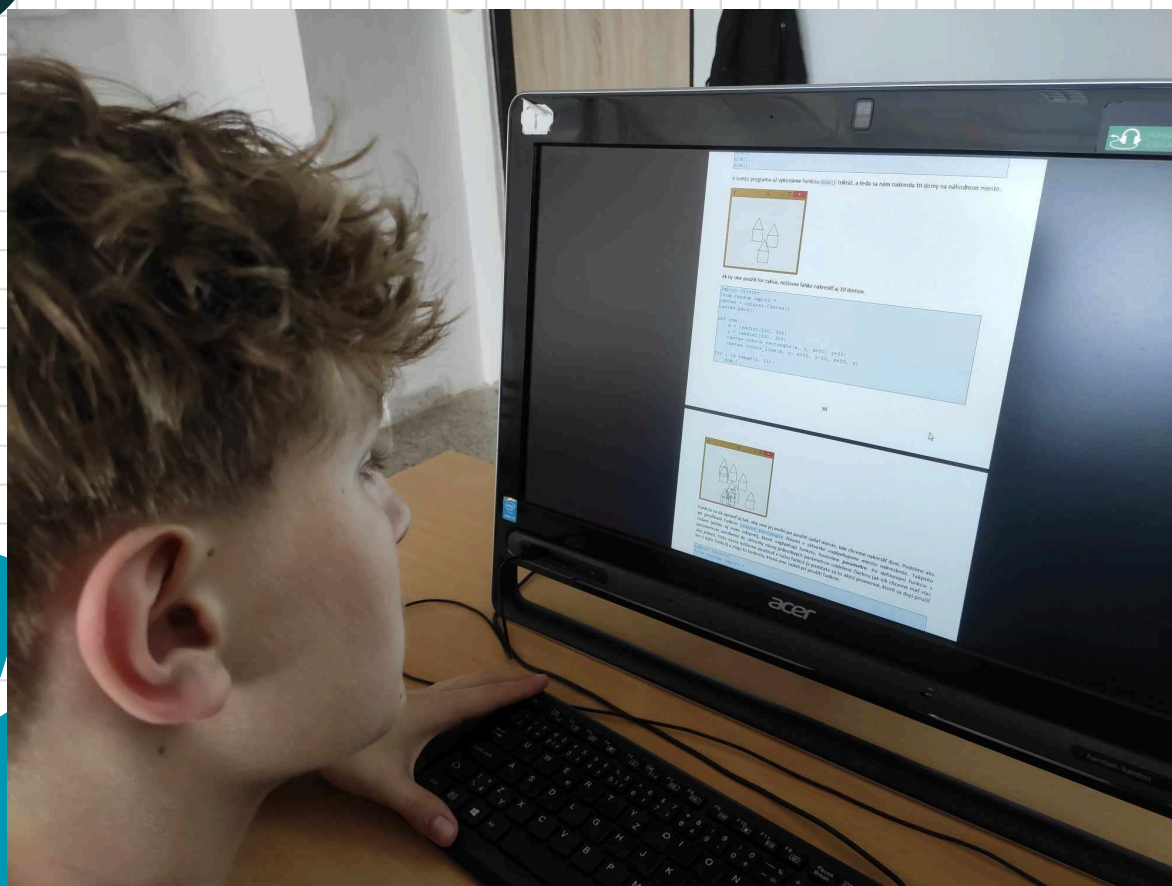
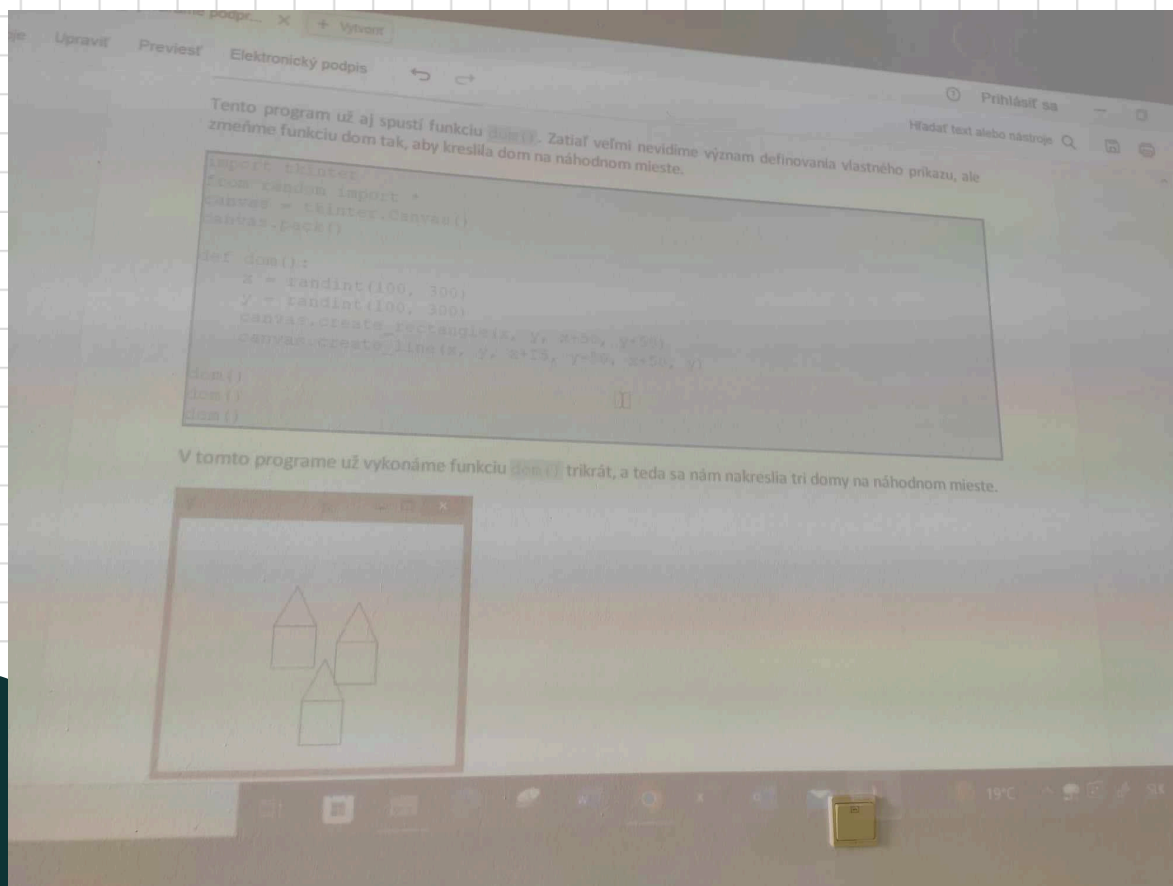




PRILOGA



FOTOGRAFIJE IZ IZVEDBE LEKCIJE

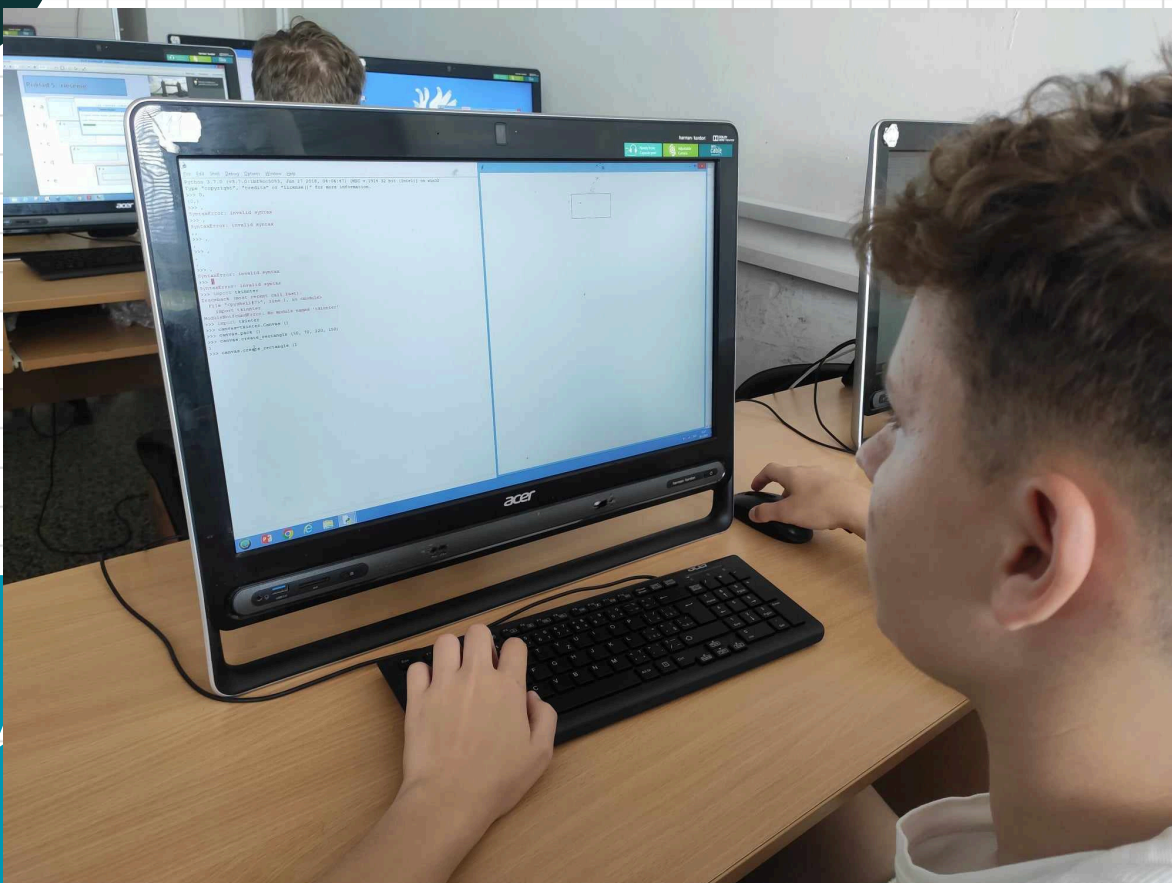
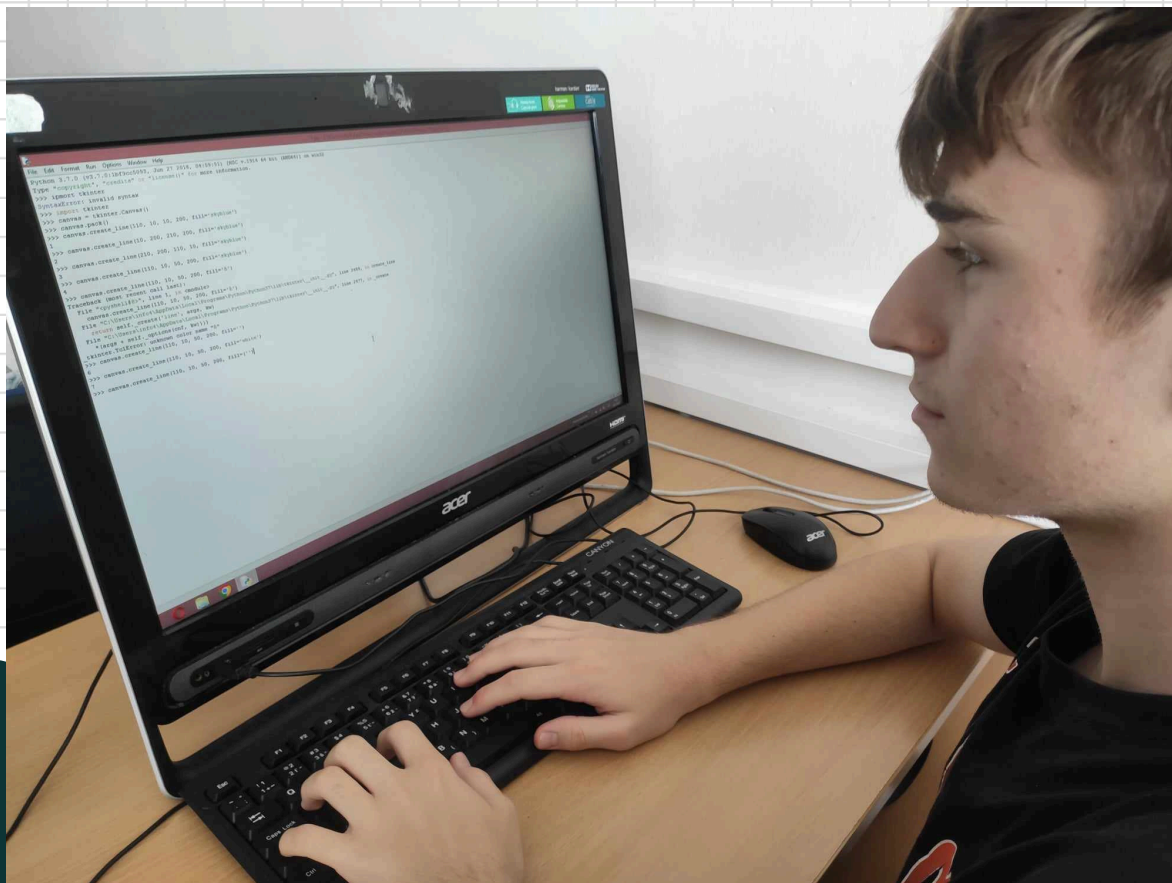




PRILOGA



FOTOGRAFIJE IZ IZVEDBE LEKCIJE





PRILOGA

ŠTUDENTSKI REZULTAT



KLICNI MEHANIZEM PONAŽARJA KLIC ZOPIS_HVIEZDICEK(30):

POVRATNI NASLOV KLICA SI ZAPOMNI

USTVARJENA JE NOVA ŠTEVILKA SPREMENLJIVKE (FORMALNI PARAMETER) IN JI JE DODELJENA VREDNOST DEJANSKEGA PARAMETRA 30

VSI STAVKI V DEFINICIJI FUNKCIJE (TELO FUNKCIJE) SE IZVEDEJO VSE SPREMENLJIVKE, KI SO BILE USTVARJENE MED IZVAJANJEM FUNKCIJE, BODO PREKLICANE

**NADZOR SE VRNE TJA, KJER JE BILA FUNKCIJA POKLICANA
NAPIŠIMO NOVO FUNKCIJO RED_CIRCLE(), KI BO IMELA DVA
PARAMETRA: KOORDINATE SREDIŠČA KROGA. FUNKCIJA NARIŠE
KROG S POLMEROM 10 IN Z DANIM SREDIŠČEM:**

DEF CERVENY_KRUH(X, Y):

**CANVAS.CREATE_OVAL(X - 10, Y - 10, X + 10, Y + 10, FILL='RED')
IN GA LAHKO IMENUJEMO NA PRIMER TAKOLE:**

UVOZI TKINTER DEF CERVENY_KRUH(X, Y):

**CANVAS.CREATE_OVAL(X - 10, Y - 10, X + 10, Y +
10, FILL='RED')**

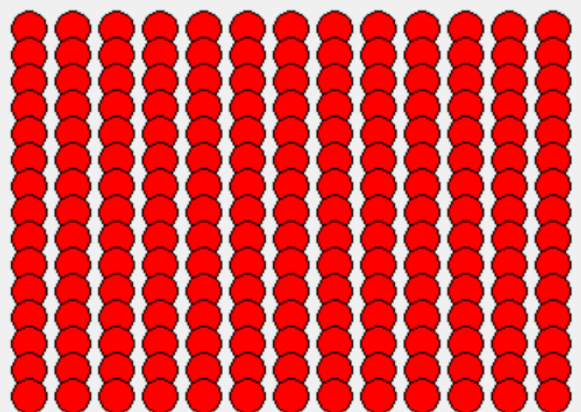
PLATNO = TKINTER.CANVAS() CANVAS.PACK()

ZA X V OBMOČJU (30, 350, 25):

ZA Y V OBSEGU (20, 240, 15):

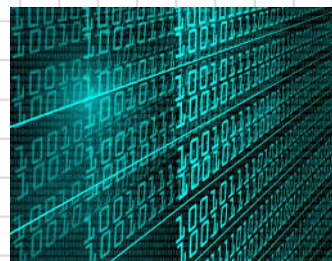
RDEČI_KROG(X, Y)

TKINTER.MAINLOOP()





NAČRT LEKCIJE



ZADEVA: PROGRAMIRANJE

TEMA LEKCIJE: PROGRAMIRANJE V PYTHONU – IZDELUJEMO PODPROGRAME.

UČNI CILJI:

- 1. RAZUMETI OSNOVNA NAČELA USTVARJANJA PODPROGRAMOV V PYTHONU Z UPORABO KNJIŽNICE TKINTER
- 2. NAUČITI SE UPORABLJATI FUNKCIJE S PARAMETRI V PYTHONU.

ORODJA:

RAČUNALNIK Z NAMEŠČENIM PYTHONOM
IN KNJIŽNICO TKINTER.
PYTHON VZORČNI UČBENIŠKI DELOVNI
LIST Z VAJAMI

HOOK/INTRO: UPORABA RAZLIČNIH BARV PRI RISANJU:

FUNKCIJA JE IMENOVANI BLOK UKAZOV
(PODPROGRAM). OPIŠEMO (DEFINIRAMO) JO
S POSEBNO KONSTRUKCIJO:

- FUNKCIJO LAHKO POKLIČEMO ŠELE, KO PYTHON ŽE POZNA NJENO DEFINICIJO

POTEK: POVEZAVA (4 MINUTE)

NAUČENA SKLADENJSKA PRAVILA

- NAČIN ZAPISA DEFINICIJE PODPROGRAMA: GLAVA IN TELO ODMIK VSEH UKAZOV V TELESU PODPROGRAMA OD LEVEGA ROBA NAČIN ZAPISA KLICA PODPROGRAMA (ZA IMENOM PODPROGRAMA PRAZEN OKLEPAJ)

POGOVOR Z UČENCI O PREDNOSTIH UPORABE PODPROGRAMOV V PYTHONU: – PODPROGRAMI LAHKO UPORABLJAJO GLOBALNO SPREMENLJIVKO CANVAS

PRI OBLIKOVANJU VNOSA PODPROGRAMA UPORABLJAMO TIPKI <TAB> IN <ENTER>

- RAZISKOVANJE (12 MINUT) – RAZISKOVANJE NOVIH UKAZOV – S POMOČJO PODPROGRAMOV SE RISANJE IZVAJA V GRAFIČNEM PODROČJU. ZDAJ BOSTE USTVARILI PODPROGRAME, KI RIŠEJO V GRAFIČNEM OKNU.

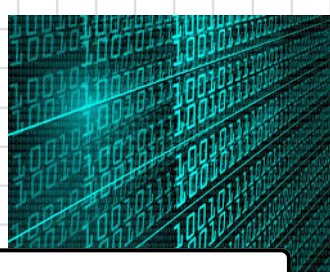
PREDSTAVITEV (15 MINUT) – POVZETEK IN RAZLAGA ZNANJA IZ POGlavJA RAZISKOVANJE PROBLEMATIKE

USTVARITE NOV PROGRAM KRESBA_PODPROGRAM.PY IN POSKUSITE:

- UVOZI TKINTER
 - PLATNO = TKINTER.CANVAS()
 - CANVAS.PACK()
 - DEF STOLI():
 - CANVAS.CREATE_RECTANGLE(10, 20, 30, 40, FILL='RDEČA')
 - STOL ()
 - V TEJ NALOGI TELO PODPROGRAMA VSEBUJE EN SAM UKAZ – NARISATI PRAVOKOTNIK.
 - ELABORAT (9 MINUT) – PODPORA ŠTUDENTOM PRI NADALJNJEM SAMOSTOJNEM USTVARJANJU PYTHON PODPROGRAMOV
 - OCENJEVANJE (5 MINUT) – REŠEVANJE TESTA ZA SAMOOCENJEVANJE, POGOVOR O ODGOVORIH.
- FAZA IZPOSTAVLJENOSTI – IZPOSTAVLJENOST NOVIH NAROČIL
– NA TABLI BOM NA KRATKO ZABELEŽIL OSNOVNE POJME



NAČRT LEKCIJE



5. DELOVNI LIST UČENCEV

1. Naloga: Ustvarite nov program `kriz.py` in v njem definirajte podprogram `kriz`, ki bo ob klicu izrisal rdeč križec:

```
uvozi tkinter
platno = tkinter.Canvas() canvas.pack() def kriz():
canvas.create_rectangle(150 - 90, 100 - 30, 150 + 90, 100 + 30, fill='rdeča') canvas.create_rectangle(150 - 30, 100 - 90,
150 + 30, 100 + 90, fill='rdeča') križ()
```

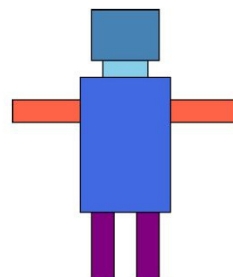
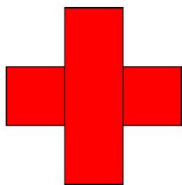
V tej nalogi telo podprograma vsebuje zaporedje dveh stavkov. Ob vsakem klicu tega podprograma se najprej nariše prvi pravokotnik, nato pa se skozenj nariše drugi pravokotnik.

2. Naloga: Ustvarite nov program `robot.py`, ki bo znal narisati robota.

V programu bodo štiri podprogrami – glava, roke, noge, telo – in vsak izmed njih bo lahko narisal delček robota. Ko jih pokličete v naslednjem vrstnem redu:

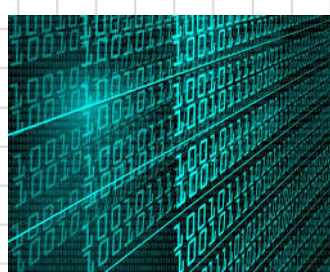
`head()` `arms()` `legs()` `body()` celoten robot je narisán.

```
uvozi tkinter
platno = tkinter.Canvas() canvas.pack() def glava():
canvas.create_rectangle(160, 40, 200, 100, fill='nebeško modra') canvas.create_rectangle(150, 10, 210, 55, fill='jekleno
modra') def telo():
canvas.create_rectangle(140, 70, 220, 190, fill='kraljevsko modra') def ruce():
canvas.create_rectangle(80, 90, 280, 110, fill='paradižnik') def nohy():
canvas.create_rectangle(150, 160, 170, 250, fill='vijolična') canvas.create_rectangle(190, 160, 210, 250, fill='vijolična')
hlava() ruce() nohy() telo()
```



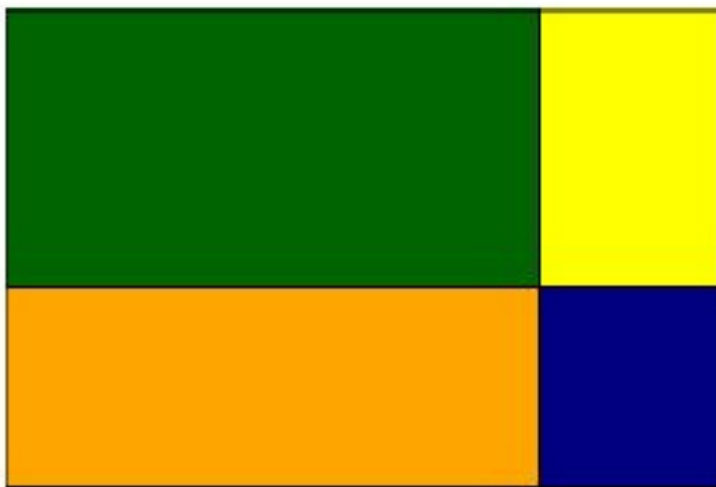


NAČRT LEKCIJE



5. DELOVNI LIST UČENCEV

3. Naloga: Ustvarite nov program `sportová_vlajka.py`, ki bo narisal športno zastavo vašega razreda. Zastava bo sestavljena iz štirih barvnih pravokotnikov, ki se dotikajo drug drugega v eni točki, kot je na spodnji sliki: Program naključno izbere koordinate x, y , ki predstavljajo stičišče vseh štirih pravokotnikov. Zunanje mere zastave naj bodo 300×200 ; določite barve na zastavi po lastni presoji.

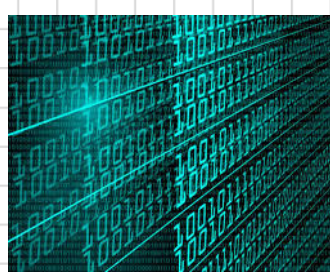


```
import tkinter import random canvas = tkinter.Canvas() canvas.pack() x = random.randint(50, 250) y = random.randint(50, 150) canvas.create_rectangle(10, 10, x, y, fill='darkgreen') canvas.create_rectangle(x, 10, 310, y, fill='yellow') canvas.create_rectangle(10, y, x, 210, fill='orange') canvas.create_rectangle(x, y, 310, 210, fill='navy')
```

Zunanja oglišča pravokotnika lahko izbiramo poljubno. V naši rešitvi ima zgornje levo oglišče koordinate $[10, 10]$ in ker bi morale biti dimenzije pravokotnika 300×200 , ima nasprotno oglišče (spodaj desno) koordinate $[310, 210]$. Notranjo naključno izbrano točko smo generirali tako, da je oddaljena vsaj 50 od robov pravokotnika, tako da bo vsako od štirih območij imelo dimenzije vsaj 50×50 .



PRILOGA

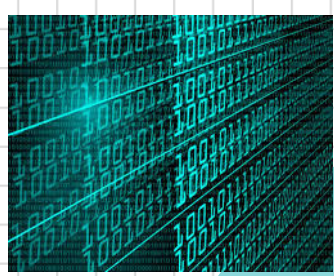


FOTOGRAFIJE IZ IZVEDBE LEKCIJE

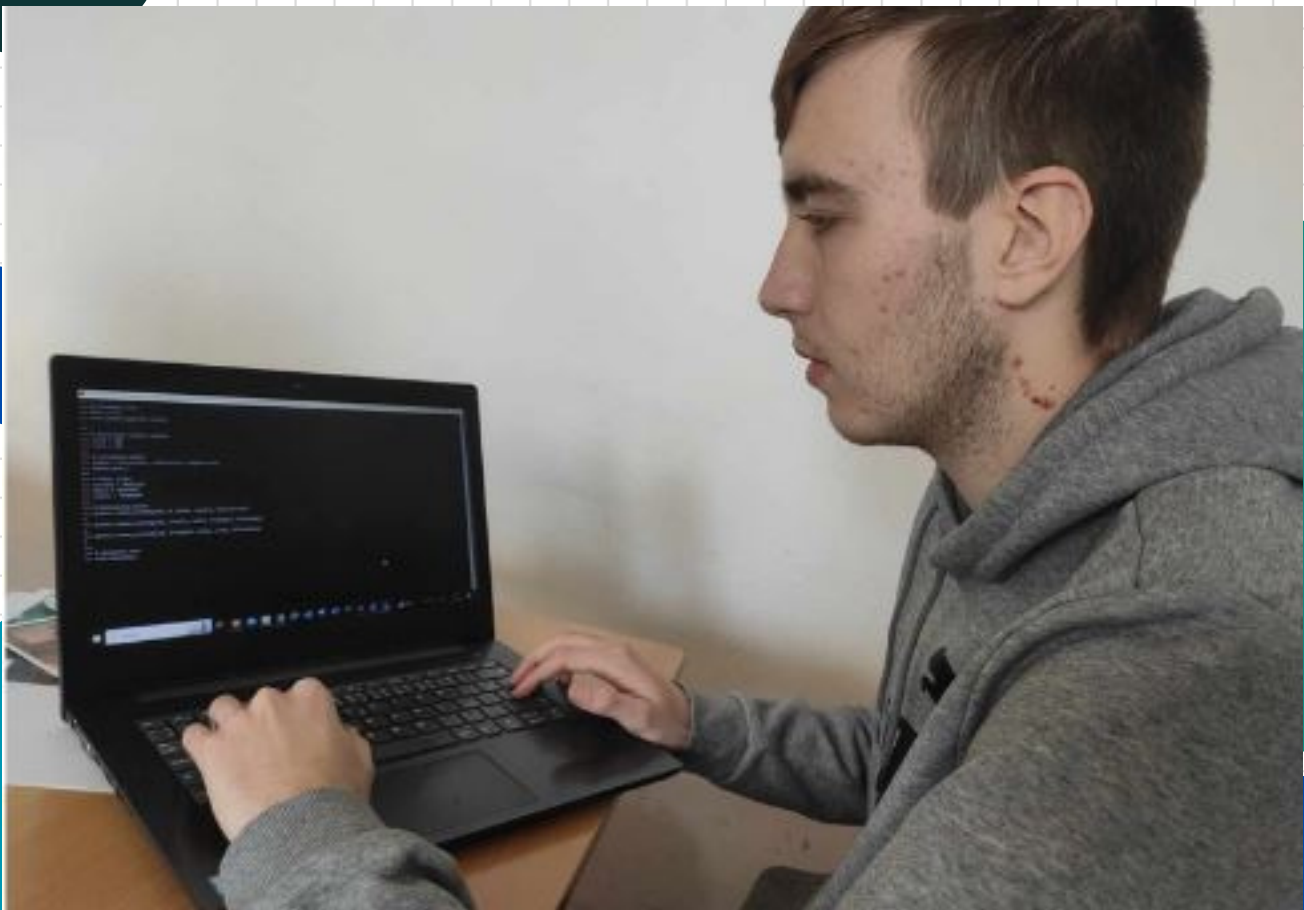




PRILOGA

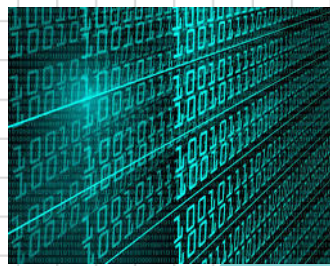


FOTOGRAFIJE IZ IZVEDBE LEKCIJE





PRILOGA



NALOGA:

SPREMENITE PROGRAM RANDOM_SQUARES.PY TAKO, DA KVADRATI NISO NARISANI SAMO NA NAKLJUČNIH POLOŽAJIH, AMPAK TUDI TAKO, DA IMA VSAK KVADRAT NAKLJUČNO VELIKOST OD INTERVALA 10 DO 100:

```
DEF RANDOM_SQUARE():
```

```
X = RANDOM.RANDINT(10, 300) Y = RANDOM.RANDINT(10, 200) A =  
RANDOM.RANDINT(10, 100) CANVAS.CREATE_RECTANGLE(X, Y, X + A, Y + A,  
FILL='ORANGE ')
```

```
IMPORT TKINTER IMPORT RANDOM CANVAS = TKINTER.CANVAS()
```

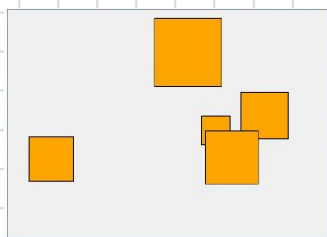
```
CANVAS.PACK() DEF RANDOM_THURSDAY():
```

```
X = RANDOM.RANDINT(10, 300) Y = RANDOM.RANDINT(10, 200) A =  
RANDOM.RANDINT(10, 100) CANVAS.CREATE_RECTANGLE(X, Y, X + A, Y + A,  
FILL='ORANGE ') DEF RANDOM_OBDLŽNIK():
```

```
X = RANDOM.RANDINT(10, 300) Y = RANDOM.RANDINT(10, 200) A =  
RANDOM.RANDINT(10, 100) B = RANDOM.RANDINT(10, 100)
```

```
CANVAS.CREATE_RECTANGLE(X, Y, X + A, Y + B, FILL='LIMETA ZELENA')
```

```
RANDOM_OBD() RANDOM_OBD() RANDOM_OBD() RANDOM_OBD() RANDOM_OBD()  
RANDOM_OBD() RANDOM_OBD() RANDOM_OBD() RANDOM_OBD() RANDOM_OBD()
```



Načrt lekcije

Serija uvodnih lekcij o programiranju v C

Tema: Kako začeti s C

Lekcija #1

Poudarek in cilji lekcije:

Ta lekcija študente seznani s programskim jezikom C, njegovo zgodovino in osnovno sintakso. Do konca te lekcije bi morali učenci znati napisati in zagnati preprost "Hello, World!" program

Potrebni materiali:

Računalniki z nameščenim prevajalnikom C (npr. GCC)
Projektor in platno
Tabla in markerji
Izročki osnovne sintakse in strukture C

Učni cilji:

Predstavite C
Predstavite Beležnico in GCC
Napišite »Hello, World!« program.

Struktura / dejavnost:

Uvod v C (15 minut)

Razprava (5 minut): Začnite s kratko zgodovino programskega jezika C. Razpravljajte o njegovem pomenu in o tem, kako je postavil temelje številnim sodobnim programskim jezikom. Označite ključne aplikacije, kot so operacijski sistemi (npr. UNIX, Linux) in vgrajeni sistemi.

Predstavitve (10 minut): S projektorjem pokažite, kje se danes uporablja jezik C, in razpravljajte o tem, zakaj je učenje C bistveno. Navedite primere priljubljene programske opreme in sistemov, napisanih v C.

Osnovna sintaksa in struktura (20 minut)

Razlaga (10 minut): Predstavite strukturo osnovnega C programa. Pojasnite vlogo `#include`, funkcije `main()` in osnovnih skladenjskih pravil. Navedite primere sintaksnih napak in kako se jim izogniti.

Primer (10 minut): Napišite preprost "Hello, World!" program na tabli ali platnu projektorja. Razčlenite vsako vrstico kode in razložite njen namen. Spodbudite učence, da postavljajo vprašanja o vsakem delu kode.

Praktična dejavnost (20 minut)

Dejavnost (15 minut): Učenci naj napišejo svoje "Hello, World!" program na svojih računalnikih. Vodite jih skozi nastavitve njihovega razvojnega okolja (po potrebi namestite prevajalnik C, kot je GCC). Zagotovite, da lahko vsak študent uspešno sestavi in izvaja svoj program.

Medvrstniški pregled (5 minut): Učence združite v pare, da pregledajo kodo drug drugega. Spodbujajte jih, naj razpravljajo o kakršnih koli težavah ali vprašanjih, s katerimi se srečujejo.

Vprašanja in odgovori ter povzetek (5 minut)

Vprašanja in odgovori (3 minute): Odprite besedo za vsa vprašanja v zvezi z lekcijo. Obravnavajte pogoste težave ali točke zmede, ki bi jih lahko imeli učenci.

Povzetek in domača naloga (2 minuti): Povzemite ključne točke lekcije. Za domačo nalogo učence prosite, naj na spletu raziščejo preprost program C, ga nekoliko spremenijo (npr. spremenijo izhodno sporočilo) in ga zaženejo.

Načrt lekcije – izroček

Serija uvodnih lekcij o programiranju v C

Tema: Kako začeti s C

Izroček #1

1. Uvod v C

Ključne točke:

C je temeljni programski jezik, ki se uporablja v številnih aplikacijah, vključno z operacijskimi sistemi in vgrajenimi sistemi.

Razumevanje jezika C pomaga pri učenju drugih programskih jezikov, kot so C++, Java in Python.

Primeri programov:

"Hello, World!": Najpreprostejši program C.

```
#include <stdio.h>

int main() {
    printf("Pozdravljen, svet!\n");
    vrni 0;
}
```

2. Osnovna sintaksa in struktura

Ključne točke:

Vsak program C mora vsebovati `#include <stdio.h>` in funkcijo `main()`.

Funkcija `main()` je začetna točka katerega koli programa C.

dejavnost:

Napišite in zaženite "Hello, World!" program.

Določite namen vsake vrstice v programu.

3. Domača naloga:

Raziščite preprost program C na spletu, ga nekoliko spremenite (npr. spremenite izhodno sporočilo) in ga zaženite.

Načrt lekcije

Serija uvodnih lekcij o programiranju v C

Tema: Spremenljivke in podatkovni tipi

Lekcija #2

Poudarek in cilji lekcije:

Poučite študente o spremenljivkah, vrstah podatkov in osnovnih vhodno/izhodnih operacijah v C.

Potrebni materiali:

Računalniki z nameščenim prevajalnikom C (npr. GCC)
Projektor in platno
Tabla in markerji
Izročki s primeri deklaracij spremenljivk in pogostih vhodno/izhodnih operacij

Učni cilji:

Poučite se o spremenljivkah/vrstah podatkov
Napišite program s printf

Struktura / dejavnost:

Pregled prejšnje lekcije (5 minut)

Razprava (5 minut): ponovite osnovno strukturo programa C in "Hello, World!" program iz prejšnje lekcije.

Poudarite pomen razumevanja sintakse, preden preidete na bolj zapletene koncepte.

Uvod v spremenljivke in podatkovne tipe (15 minut)

Razlaga (10 minut): Predstavite koncept spremenljivk kot ograda za podatke. Pojasnite razliko med različnimi vrstami podatkov (npr. int, float, char, double) in njihovo tipično uporabo.

Primer (5 minut): Napišite preprost program, ki deklarira različne tipe spremenljivk in uporablja printf za izpis njihovih vrednosti. Pokažite, kako dodeljujete vrednosti spremenljivkam in izvajate osnovne aritmetične operacije.

Praktična dejavnost št. 1: Deklaracija spremenljivk (15 minut)

Dejavnost (10 minut): Učenci naj v novem programu deklarirajo spremenljivke različnih podatkovnih tipov.

Prosimo jih, naj izvedejo osnovne operacije, kot sta seštevanje in množenje, ter natisnejo rezultate.

Medsebojni pregled (5 minut): Učenci izmenjajo svoje programe in preverijo, ali so deklaracije in operacije spremenljivk pravilne. Spodbujajte jih, naj razpravljajo o morebitnih razlikah ali napakah.

Uvod v osnovni vhod/izhod (10 minut)

Pojasnilo (5 minut): Predstavite funkcijo scanf za sprejemanje uporabniških vnosov. Pojasnite, kako deluje v povezavi s printf za ustvarjanje interaktivnih programov.

Primer (5 minut): Napišite program, ki uporabnika vpraša po starosti in mu to nato natisne nazaj z uporabo scanf in printf.

Praktična dejavnost št. 2: Interaktivni program (15 minut)

Dejavnost (10 minut): Učenci bodo spremenili svoj prejšnji program, da bo vključeval uporabniški vnos. Na primer, prosite uporabnika za dve številki, izvedite operacijo z njima (na primer seštevanje) in natisnite rezultat.

Podpora (5 minut): Razširite sobo, da pomagate učencem pri sintakasnih ali logičnih napakah, na katere naletijo.

Vprašanja in odgovori ter povzetek (5 minut)

Vprašanja in odgovori (3 minute): študentom omogočite, da postavljajo vprašanja o spremenljivkah, tipih podatkov ali vhodno/izhodnih operacijah.

Povzetek in domača naloga (2 minuti): Povzemite lekcijo. Dodelite domačo nalogo, kjer učenci ustvarijo program, ki sprejme uporabniške vnose za tri različne tipe podatkov in jih izpiše v oblikovanem nizu.

Načrt lekcije – izroček

Serija uvodnih lekcij o programiranju v C

Tema: Spremenljivke in podatkovni tipi

Izroček št. 2

1. Uvod v spremenljivke

Ključne točke:

Spremenljivke so ograde za podatke. V jeziku C ima vsaka spremenljivka določen tip podatkov.

Pogosti tipi podatkov: int (celo število), float (število s plavajočo vejico), char (znak), double (plavajoča vejica z dvojno natančnostjo).

Primer kode:

```
int starost = 25;
višina plovca = 5,9;
char začetnica = 'A';
dvojna razdalja = 12345,67;
```

2. Vhod in izhod

Ključne točke:

Uporabite printf za izpis podatkov in scanf za vnos.

Primer kode:

```
int starost;
printf("Vnesite svojo starost: ");
scanf("%d", &starost);
printf("Vnesli ste: %d\n", starost);
```

3. Domača naloga:

Ustvarite program, ki sprejme uporabniški vnos za tri različne tipe podatkov in jih izpiše v oblikovanem nizu.

Načrt lekcije

Serija uvodnih lekcij o programiranju v C

Tema: Nadzorne strukture

Lekcija #3

Poudarek in cilji lekcije:

Ta lekcija študente seznani s krmilnimi strukturami v C, vključno s stavki if-else in zankami. Do konca lekcije bi morali biti učenci sposobni uporabljati te strukture za nadzor poteka svojih programov.

Potrebni materiali:

Računalniki z nameščenim prevajalnikom C (npr. GCC)
Projektor in platno
Tabla in markerji
Izročki s primeri stavkov if-else in zank

Učni cilji:

Spoznajte osnovne vhode/izhode
Spoznajte stavke if/if-else

Struktura / dejavnost:

Pregled prejšnje lekcije (5 minut)

Razprava (5 minut): ponovite koncepte spremenljivk, podatkovnih tipov in osnovnega vnosa/izhoda. Poudarite, kako se bodo ti elementi uporabljali v povezavi z nadzornimi strukturami.

Uvod v stavke If-Else (15 minut)

Razlaga (10 minut): Pojasnite sintakso in uporabo stavkov if-else. Razpravljajte o konceptu pogojne logike in o tem, kako programom omogoča sprejemanje odločitev.

Primer (5 minut): Napišite program, ki od uporabnika zahteva število in izpiše, ali je sodo ali liho. Pojasnite, kako se stavek if-else uporablja za preverjanje pogojev.

Praktična dejavnost št. 1: Pogojna logika (15 minut)

Dejavnost (10 minut): Učenci napišejo program, ki sprejema uporabniški vnos in uporablja stavke if-else za sprejemanje odločitev (npr. preverjanje, ali je število pozitivno, negativno ali nič).

Medsebojni pregled (5 minut): študenti naj izmenjajo svoje programe in pregledajo uporabljeno logiko.

Spodbujajte jih, da preizkusijo programe drug drugega z različnimi vhodnimi podatki.

Uvod v zanke (15 minut)

Pojasnilo (10 minut): Predstavite zanki for in while. Pojasnite, kako se zanke uporabljajo za ponavljanje dejanj v programu, in razpravljajte o scenarijih, kjer je vsaka vrsta zanke primerna.

Primer (5 minut): Napišite program, ki izpisuje števila od 1 do 10 z uporabo zanke for. Nato ga prepišite z uporabo zanke while.

Praktična dejavnost št. 2: Strukture zanke (15 minut)

Dejavnost (10 minut): Učenci napišejo program, ki uporablja zanko za izračun faktoriala števila ali ustvarjanje tabele množenja.

Podpora (5 minut): Pomagajte študentom pri odpravljanju napak v njihovih zankah in zagotavljanju pravilne logike.

Vprašanja in odgovori ter povzetek (5 minut)

Vprašanja in odgovori (3 minute): Odgovorite na vsa vprašanja o nadzornih strukturah. Razpravljajte o pogostih napakah, kot so neskončne zanke ali neustrezna uporaba if-else.

Povzetek in domača naloga (2 minuti): Povzemite lekcijo. Dodelite domačo nalogo, kjer učenci napišejo program z uporabo stavkov if-else in zank, da ustvarijo preprost kalkulator.

Načrt lekcije – izroček

Serija uvodnih lekcij o programiranju v C

Tema: Nadzorne strukture

Izroček št. 3

1. Stavki If-Else

Ključne točke:

Stavki if-else omogočajo programu sprejemanje odločitev na podlagi pogojev.

Primer kode:

```
int število;  
printf("Vnesite številko: ");  
scanf("%d", &število);  
  
if (število % 2 == 0) { printf("Število je  
sodo.\n");  
} else { printf("Število je liho.\n");  
}
```

2. Zanke

Ključne točke:

Zanke omogočajo programu ponavljanje dejanj.

Običajni zanki sta for in while.

Primer kode:

```
for (int i = 1; i <= 10; i++) { printf("%d\n", i);  
}
```

3. Domača naloga:

Napišite program z uporabo stavkov if-else in zank, da ustvarite preprost kalkulator.

Vaše besedilo odstavka

Načrt lekcije

Serija uvodnih lekcij o programiranju v C

Tema: Funkcije

Lekcija št. 4

Poudarek in cilji lekcije:

Ta lekcija učence pouči o funkcijah v C, njihovi pomembnosti in o tem, kako jih ustvariti in uporabljati. Do konca lekcije bi morali biti učenci sposobni definirati in priklicati osnovne funkcije v svojih programih.

Potrebni materiali:

Računalniki z nameščenim prevajalnikom C (npr. GCC)
Projektor in platno
Tabla in markerji

- Izročki s primeri definicij funkcij in klicev funkcij

Učni cilji:

Naučite se definirati osnovne funkcije v C
Hands-on practice

Struktura / dejavnost:

Pregled prejšnje lekcije (5 minut)

Razprava (5 minut): Hitro preglejte kontrolne strukture in poudarite, kako bodo funkcije pomagale organizirati in ponovno uporabiti kodo v kompleksnejših programih.

Uvod v funkcije (15 minut)

Razlaga (10 minut): Definirajte funkcije in razložite namen razdelitve kode na manjše bloke, ki jih je mogoče ponovno uporabiti. Razpravljajte o sintaksi funkcije, vključno z vrnjenimi vrstami, imeni funkcij in parametri.

Primer (5 minut): Napišite preprosto funkcijo, ki izračuna kvadrat števila. Pojasnite, kako je funkcija definirana, poklicana in kako vrne vrednost.

Praktična dejavnost št. 1: Funkcije pisanja (15 minut)

Dejavnost (10 minut): Učenci naj napišejo lastno funkcijo, ki opravlja določeno nalogo, kot je pretvorba Celzija v Fahrenheite. Prepričajte se, da razumejo, kako posredovati parametre in vrniti vrednosti.

Medsebojni pregled (5 minut): Učenci izmenjujejo svoje programe in preizkušajo funkcije drug drugega.

Razširitev funkcij (15 minut)

Razlaga (10 minut): Pogovorite se o uporabi več funkcij v enem samem programu. Pojasnite, kako lahko funkcije kličejo druge funkcije in kako upravljati odvisnosti funkcij.

Primer (5 minut): Napišite program, ki vključuje dve funkciji: eno za preverjanje vnosa in drugo za izvajanje izračuna. Pokažite, kako te funkcije medsebojno delujejo.

Praktična dejavnost št. 2: Interakcija funkcij (15 minut)

Dejavnost (10 minut): Učenci bodo ustvarili program, ki uporablja vsaj dve funkciji, ki delujeta skupaj, kot je preprost kalkulator, ki potrdi vnos pred izvedbo operacij.

Podpora (5 minut): Zagotovite smernice za odpravljanje napak in razumevanje obsegov funkcij.

Vprašanja in odgovori ter povzetek (5 minut)

Vprašanja in odgovori (3 minute): Odgovorite na vsa vprašanja v zvezi s funkcijami. Razpravljajte o pogostih težavah, kot so preobremenitev funkcij in spori pri poimenovanju.

Povzetek in domača naloga (2 minuti): Povzemite lekcijo. Za domačo nalogo prosite študente, naj spremenijo program, da bo vključeval funkcije za vnos, obdelavo in izhod.

Načrt lekcije – izroček

Serija uvodnih lekcij o programiranju v C

Tema: Funkcije

Izroček št. 4

1. Uvod v funkcije

Ključne točke:

Funkcije so ponovno uporabni bloki kode, ki izvajajo določene naloge.

Funkcije imajo vrnjeno vrsto, ime in parametre.

Primer kode:

```
int square(int num) { return num * num;
}

int main() { int rezultat = kvadrat(5);
    printf("Kvadrat 5 je %d\n", rezultat);
    vrni 0;
}
```

2. Funkcije pisanja in klicanja

dejavnost:

Napišite funkcijo, ki pretvori Celzija v Fahrenheite in jo pokliče v `main()`.

3. Domača naloga:

Spremenite program tako, da vključuje funkcije za vnos, obdelavo in izhod.

Načrt lekcije

Serija uvodnih lekcij o programiranju v C

Tema: Nizi in nizi

Lekcija #5

Poudarek in cilji lekcije:

Ta lekcija učencem predstavi matrike in nize v C ter razloži, kako jih deklarirati, inicializirati in uporabiti v programih. Do konca lekcije bi morali učenci znati manipulirati z nizi in nizi v osnovnih programih C.

Potrebni materiali:

Računalniki z nameščenim prevajalnikom C (npr. GCC)
Projektor in platno
Tabla in markerji

- Izročki s primeri operacij z nizi in nizi

Učni cilji:

Predstavitev matrik in nizov Praktična praksa

Struktura / dejavnost:

Pregled prejšnje lekcije (5 minut)

Razprava (5 minut): ponovite koncept funkcij in njihovo uporabo v C. Poudarite, kako lahko funkcije poenostavijo delo z nizi in nizi.

Uvod v polja (15 minut)

Pojasnilo (10 minut): Pojasnite, kaj so polja, vključno z njihovo sintakso in kako jih deklarirati in inicializirati. Razpravljajte o različnih vrstah nizov, kot so enodimenzionalni in večdimenzionalni nizi.

Primer (5 minut): Napišite program, ki uporablja matriko za shranjevanje in seštevanje seznama števil.

Razložite vsak korak programa, od deklaracije do ponovitve in seštevanja.

Praktična dejavnost št. 1: Manipulacija nizov (15 minut)

Dejavnost (10 minut): Učenci napišejo program, ki uporablja matriko za shranjevanje uporabniških vnosov (npr. seznam ocen) in nato izračuna povprečje.

Podpora (5 minut): Pomagajte študentom pri odpravljanju napak, kot so napake izven meja polja in nepravilno indeksiranje.

Uvod v godala (15 minut)

Razlaga (10 minut): Pogovorite se o tem, kako so nizi predstavljeni v C z uporabo nizov znakov. Pojasnite, kako deklarirati, inicializirati in manipulirati z nizi.

Primer (5 minut): Napišite program, ki sprejme vnos niza od uporabnika in ga natisne obratno. Učence vodite skozi logiko manipulacije nizov.

Praktična dejavnost št. 2: Operacije nizov (15 minut)

Dejavnost (10 minut): Učenci napišejo program, ki zahteva ime uporabnika in nato izvede različne operacije, kot je štetje števila znakov ali iskanje prvega samoglasnika.

Podpora (5 minut): nudi pomoč pri težavah s prekinitvijo nizov in pogostih napakah pri manipulaciji nizov.

Vprašanja in odgovori ter povzetek (5 minut)

Vprašanja in odgovori (3 minute): študentom omogočite, da postavljajo vprašanja o nizih in nizih.

Razpravljajte o pomembnosti razumevanja teh konceptov za naprednejše programiranje.

Povzetek in domača naloga (2 minuti): Povzemite lekcijo. Dodelite domačo nalogo, pri kateri učenci napišejo program, ki uporablja nize in nize za upravljanje preprostega seznama stikov.

Načrt lekcije – izroček

Serija uvodnih lekcij o programiranju v C

Tema: Nizi in nizi

Izroček št. 5

1. Uvod v nize

Ključne točke:

Nizi so zbirke spremenljivk istega tipa.

Nizi so indeksirani z nič; prvi element je pri indeksu 0.

Primer kode:

```
int števila [5] = {1, 2, 3, 4, 5};  
printf("Prva številka: %d\n", številke[0]);
```

2. Uvod v strune

Ključne točke:

Nizi v C so nizi znakov, ki se končajo z ničelnim znakom `\0`.

Primer kode:

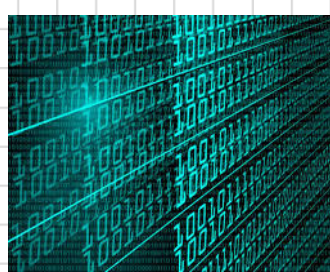
```
char ime [20];  
printf("Vnesite svoje ime: ");  
scanf("%s", ime);  
printf("Pozdravljeni, %s!\n", ime);
```

3. Domača naloga:

Napišite program, ki uporablja nize in nize za upravljanje preprostega seznama stikov.



UČNA PRIPRAVA



PREDMET: PROGRAMIRANJE IN KODIRANJE

TEMA: UVODNE LEKCIJE O PROGRAMSKEM JEZIKU PYTHON – LEKCIJA 1

POUDAREK IN CILJI LEKCIJE:

1. TA LEKCIJA ŠTUDENTE SEZNANI Z OSNOVAMI PROGRAMSKEGA JEZIKA PYTHON, VKLJUČNO Z NJEGOVO ZGODOVINO, JEZIKOVNIMI ZNAČILNOSTMI IN PISANJEM PRVEGA PROGRAMA PYTHON.
2. DO KONCA TE LEKCIJE BI MORALI BITI UČENCI SPOSOBNI RAZUMETI OSNOVNO SINTAKSO JEZIKA PYTHON TER BITI SPOSOBNI ZGRADITI IN ZAGNATI PREPROST "HELLO, WORLD!" PROGRAM.

POTREBNI MATERIALI:

- RAČUNALNIKI Z NAMEŠČENIM PYTHONOM (NPR. VS CODE, PYCHARM ALI DRUGO RAZVOJNO OKOLJE)
- PROJEKTOR IN PLATNO
- TABLA IN MARKERJI
- OSNOVNA SINTAKSA PYTHON IN DODELITVE STRUKTURE

1. UVOD V PYTHON (15 MINUT)

POGOVOR (5 MINUT): ZAČNITE S KRATKO ZGODOVINO PROGRAMSKEGA JEZIKA PYTHON, KI GA JE RAZVIL GUIDO VAN ROSSUM V POZNIH OSEMDESETIH LETIH. POGOVORITE SE O RAZVOJU JEZIKA IN O TEM, KAKO JE POSTAL PRILJUBLJEN ZARADI ENOSTAVNOSTI UČENJA IN ŠIROKE UPORABE. POUČITE, KAKO JE PYTHON POSTAVIL TEMELJE ZA SODOBNE PROGRAMSKE JEZIKE IN KAKO JE ŠE POSEBEJ PRILJUBLJEN V ZNANSTVENEM RAČUNALNIŠTVU, SPLETNEM RAZVOJU IN OBDELAVI PODATKOV.

PREDSTAVITEV (10 MINUT): OPIŠITE TRENUTNO UPORABO PYTHONA V RAZLIČNIH PANOGAH. POGOVORITE SE O TEM, ZAKAJ BI SE MORALI NAUČITI PYTHON, IN POKAŽITE PRIMERE PRILJUBLJENIH APLIKACIJ, KI TEMELJIJO NA PYTHONU, KOT JE ANALIZA PODATKOV ZA YOUTUBE, INSTAGRAM IN NASA.

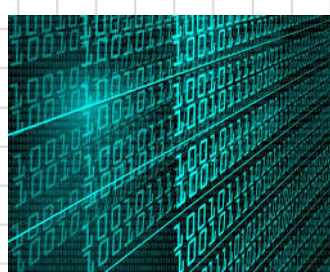
2. OSNOVNA SINTAKSA IN STRUKTURA (20 MINUT)

POJASNILO (10 MINUT): POKAŽI STRUKTURO PREPROSTEGA PROGRAMA PYTHON. POJASNITE UPORABO UVOZA, VLOGO FUNKCIJE MAIN() IN SINTAKTIČNE FUNKCIJE PYTHONA, KOT JE POMEN ZAMIKA. POJASNITE, KAKO PYTHON OBRAVNAVA IZRAZE IN KAKŠNE NAPAKE LAHKO PRIDE. OPOZORITE NA ENOSTAVNE NAPAKE, KOT JE NA PRIMER NEZAMIK, IN POJASNITE, KAKO SE JIM IZOGNITI.

PRIMER (10 MINUT): NAPIŠITE "HELLO, WORLD!" PROGRAM NA TABLI ALI PLATNU PROJEKTORJA. PODROBNO RAZLOŽITE VLOGO VSAKE VRSTICE KODE, VKLJUČNO S TEM, KAKO DELUJE FUNKCIJA PRINT(), RAVNANJEM Z NIZI IN RAZUMEVANJEM IZHODNEGA REZULTATA. SPODBUDITE UČENCE, DA POSTAVLJAJO VPRAŠANJA O VSAKEM DELU KODE.



UČNA PRIPRAVA



3. PRAKTIČNA AKTIVNOST (20 MINUT)

DEJAVNOST (15 MINUT): UČENCE PROSITE, NAJ NAPIŠEJO SVOJE "HELLO, WORLD!" NJIHOV PROGRAM V PYTHONU. POMAGAJTE JIM PRI NASTAVITVI NJIHOVEGA RAZVOJNEGA OKOLJA (NA PRIMER Z UPORABO IDLE ALI PYCHARM) IN POSKRBITI, DA LAHKO VSI USPEŠNO IZVAJAJO SVOJ PROGRAM. PREPRIČAJTE SE, DA VSI UČENCI RAZUMEJO, KAKO KODA DELUJE IN KAKO KRMARITI PO OKOLJU PYTHON.

MEDVRSTNIŠKI PREGLED (5 MINUT): UČENCE ZDRUŽITE V PARE, DA PREGLEDAJO KODO DRUG DRUGEGA. SPODBUJAJTE JIH, DA RAZPRAVLJAJO O KAKRŠNIH KOLI VPRAŠANJIH ALI TEŽAVAH, KI SE POJAVIJO, IN DRUG DRUGEMU POMAGAJO IZBOLJŠATI KODO.

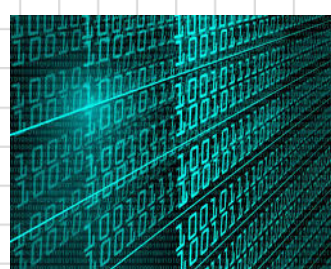
4. VPRAŠANJA IN ODGOVORI TER POVZETEK (5 MINUT)

VPRAŠANJA IN ODGOVORI (3 MINUTE): ODPRITE BESEDO, DA LAHKO UČENCI POSTAVLJAJO VPRAŠANJA. POGOVORITE SE O POGOSTIH TEŽAVAH ALI TOČKAH ZMEDE, KI SO SE MORDA POJAVILE MED LEKCIJO. UČENCEM OMOGOČITE, DA PODAJO POV RATNE INFORMACIJE O SVOJEM RAZUMEVANJU LEKCIJE.

POVZETEK IN DOMAČA NALOGA (2 MINUTI): POVZEMITE GLAVNE TOČKE LEKCIJE, POUDARITE OSNOVNO SINTAKSO PYTHON IN "HELLO, WORLD!" POMEMBNO ST PROGRAM A. KOT DOMAČO NALOGO UČENCE PROSITE, NAJ NA SPLETU NAJDEJO PREPROST PROGRAM PYTHON, SPREMENIJO NJEGOVO IZHODNO SPOROČILO IN GA ZAŽENEJO. PROSITE JIH, NAJ PRINESEJO SPREMENJENO KODO V NASLEDNJI RAZRED IN SE PRIPRAVIJO NA PREDSTAVITEV.



UČNA PRIPRAVA



PREDMET: PROGRAMIRANJE IN KODIRANJE

TEMA: UVODNE LEKCIJE O PROGRAMSKEM JEZIKU PYTHON - LEKCIJA 1

1. UVOD V PYTHON

GLAVNE TOČKE:

PYTHON JE PREPROST PROGRAMSKI JEZIK NA VISOKI RAVNI, KI SE UPORABLJA NA ŠTEVILNIH PODROČJIH, KOT SO SPLETNI RAZVOJ, OBDELAVA PODATKOV, ZNANSTVENO RAČUNALNIŠTVO IN UMETNA INTELIGENCA.

RAZVOJ PYTHONA SEGA V POZNA OSEMDESETA LETA PREJŠNJEGA STOLETJA, KO JE GUIDO VAN ROSSUM USTVARIL JEZIK ZA LAŽJE UČENJE IN BERLJIVOST.

OBIČAJNE APLIKACIJE PYTHONA VKLJUČUJEJO PRILJUBLJENE PLATFORME, KOT SO YOUTUBE, INSTAGRAM IN PODATKOVNA ANALITIKA NASA.

PRIMER PROGRAMA:

"POZDRAVLJEN, SVETI!" – EDEN NAJPREPROSTEJŠIH PROGRAMOV PYTHON:

```
PRINT("POZDRAVLJEN, SVETI!")
```

2. OSNOVNA SINTAKSA IN STRUKTURA

GLAVNE TOČKE:

V PYTHONU JE STRUKTURA KODE MOČNO ODVISNA OD ZAMIKA. ZAMIK DOLOČA BLOKE KODE, ZATO JE KLJUČNEGA POMENA, DA GA PRAVILNO UPORABITE.

FUNKCIJA PRINT() JE ENO NAJOSNOVNEJŠIH ORODIJ V PYTHONU, KI NAM OMOGOČA PRIKAZ BESEDILA IN DRUGIH PODATKOV NA ZASLONU.

PYTHON NE ZAHTEVA PODPIČJA NA KONCU VRSTIC, ZA RAZLIKO OD MNOGIH DRUGIH PROGRAMSKIH JEZIKOV.

DEJAVNOST:

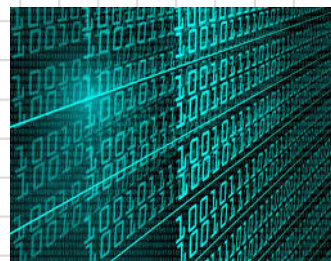
NAPIŠITE IN ZAŽENITE "HELLO, WORLD!" PROGRAM V LASTNEM RAZVOJNEM OKOLJU. DOLOČITE NAMEN VSAKE VRSTICE PROGRAMA IN SE PREPRIČAJTE, DA RAZUMETE, KAKO PROGRAM DELUJE.

3. DOMAČA NALOGA

USTVARITE PROGRAM PYTHON, KI SE GLASI "HELLO, WORLD!" PRIKAŽE OSEBNO SPOROČILO (NA PRIMER VAŠE IME ALI NAJLJUBŠI CITAT) NAMESTO SPOROČILA. POSKUSITE EKSPERIMENTIRATI S FUNKCIJO PRINT(), NA PRIMER Z UPORABO RAZLIČNIH NAREKOV ALI DODAJANJEM NOVE VRSTICE V PROGRAM, KI PRIKAŽE DODATNO BESEDILO. SHRANITE SPREMENJEN PROGRAM IN GA PRINESITE V NASLEDNJI RAZRED, DA GA PREDSTAVITE RAZREDU.



UČNA PRIPRAVA



PREDMET: PROGRAMIRANJE IN KODIRANJE

TEMA: VREDNOSTI IN TIPI V PROGRAMSKEM JEZIKU PYTHON – LEKCIJA 2

POUDARKI IN CILJI LEKCIJE:

UVOD V TEMELJNE TIPE PODATKOV, KI SE UPORABLJAJO V PYTHONU, VKLJUČNO S CELIMI ŠTEVILI, DECIMALNIMI VREDNOSTMI, NIZI IN LOGIČNIMI VREDNOSTMI.

DO KONCA LEKCIJE BI MORALI UČENCI ZNATI PREPOZNATI IN PRAVILNO UPORABITI OSNOVNE TIPE PODATKOV TER RAZUMETI, KAKO OPERACIJE VPLIVAJO NA TIP TEH VREDNOSTI.

POTREBNI MATERIALI:

- RAČUNALNIKI Z NAMEŠČENIM PYTHONOM (NPR. VISUAL STUDIO CODE, PYCHARM ALI DRUGO RAZVOJNO OKOLJE)
- PROJEKTOR ALI ZASLON
- TABELA IN OZNAČEVALCI
- PRILOGE, KI PRIKAZUJEJO KATEGORIJE PODATKOV

UVOD V VREDNOSTI IN VRSTE (15 MINUT)

RAZPRAVA (5 MINUT): ZAČNITE S KRATKO RAZPRAVO O POMENU VREDNOSTI V PROGRAMIRANJU IN POMEMBNOSTI PODATKOVNIH TIPOV. RAZPRAVLJAJTE O TEM, KAKO VREDNOSTI PREDSTAVLJAJO PODATKE V PROGRAMIH TER KAKO PODATKOVNI TIPI DOLOČAJO NAČIN OBDELAVE TEH PODATKOV.

PREDSTAVITEV (10 MINUT): PREDSTAVITEV RAZLIČNIH TIPOV PODATKOV, VKLJUČNO S CELIMI ŠTEVILI, ŠTEVILI S PLAVAJOČO VEJICO, NIZI IN LOGIČNIMI VREDNOSTMI. RAZPRAVLJAJTE O KLJUČNIH ZNAČILNOSTIH IN PRIMERIH UPORABE VSAKEGA TIPA. PRIKAŽITE PRIMERE, KAKO SE TE VRSTE PODATKOV UPORABLJAJO V RESNIČNEM SVETU, NA PRIMER PRI MATEMATIČNIH OPERACIJAH, OBDELAVI BESEDIL IN POGOJNI LOGIKI.

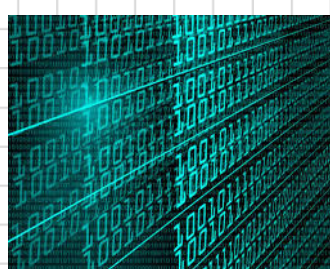
OSNOVNA SINTAKSA IN STRUKTURA (20 MINUT)

POJASNILO (10 MINUT): PRIKAZATI, KAKO JE MOGOČE DEFINIRATI IN UPORABITI OSNOVNE VRSTE PODATKOV V PYTHONU. RAZLOŽITI, KAKO SE REZULTAT IZRAZA SPREMINJA GLEDE NA UPORABLJENI PODATKOVNI TIP. NA PRIMER, KAKO SEŠTEVANJE CELEGA ŠTEVILA IN ŠTEVILA S PLAVAJOČO VEJICO USTVARI REZULTAT TER KAJ SE ZGODI, KO POSKUŠAMO ZDRUŽITI CELO ŠTEVILO IN BESEDILO.

PRIMER (10 MINUT): PRIPRAVITE PREPROSTE PRIMERE UPORABE RAZLIČNIH VRST VREDNOSTI NA TABLI ALI PLATNU PROJEKTORJA. PRIKAŽITE, KAKO SEŠTEVATI DVE ŠTEVILKI, KAKO ZDRUŽITI DVE BESEDILI IN KAKO UPORABITI LOGIČNE OPERATORJE ZA DOLOČANJE POGOJEV. POJASNITE, DA JE PRAVILNA UPORABA VREDNOSTI IN TIPOV KLJUČNA ZA UČINKOVITO DELOVANJE PROGRAMOV.



UČNA PRIPRAVA



3. PRAKTIČNA DEJAVNOST (20 MINUT)

DEJAVNOST (15 MINUT): UČENCI NAJ NAPIŠEJO PREPROSTO KODO V PYTHONU, KI VKLJUČUJE RAZLIČNE VRSTE PODATKOV, KOT SO CELA ŠTEVILA, ŠTEVILA S PLAVAJOČO VEJICO, NIZI IN LOGIČNE VREDNOSTI. POMAGAJTE JIM PRI VZPOSTAVITVI RAZVOJNEGA OKOLJA IN ZAGOTOVITE, DA LAHKO VSI USPEŠNO IZVAJAJO SVOJE PROGRAME. PREVERITE, ALI UČENCI RAZUMEJO REZULTATE IN RAZLIKE MED POSAMEZNIMI VRSTAMI.

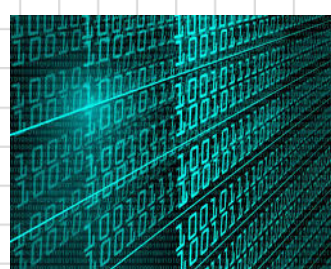
4. VPRAŠANJA IN ODGOVORI TER POVZETEK (5 MINUT)

VPRAŠANJA IN ODGOVORI (3 MINUTE): ODPRITE RAZPRAVO, DA UČENCI LAHKO POSTAVLJAJO VPRAŠANJA. OBRAVNAVAJTE POGOSTE TEŽAVE ALI TOČKE ZMEDE, KI SO SE MORDA POJAVILE MED LEKCIJO. UČENCEM OMOGOČITE, DA IZRAZIJO SVOJE MNENJE O RAZUMEVANJU LEKCIJE.

POVZETEK IN DOMAČA NALOGA (2 MINUTI): POVZEMITE KLJUČNE TOČKE LEKCIJE IN IZPOSTAVITE POMEN RAZLIČNIH TIPOV PODATKOV TER NJIHOVO PRAVILNO UPORABO. KOT DOMAČO NALOGO UČENCE PROSITE, NAJ USTVARIJO PREPROST PROGRAM V PYTHONU, KI UPORABLJA RAZLIČNE VRSTE VREDNOSTI. NA PRIMER, PROSITE JIH, NAJ NAPIŠEJO PROGRAM, KI IZRAČUNA VSOTO DVEH ŠTEVIL IN NATO PRIKAŽE SPOROČILO Z REZULTATOM.



UČNA PRIPRAVA



PREDMET: PROGRAMIRANJE IN KODIRANJE

TEMA: VREDNOSTI IN TIPI V PROGRAMSKEM JEZIKU PYTHON - LEKCIJA 2

1. VREDNOSTI IN KATEGORIJE

GLAVNE TOČKE:

VREDNOSTI SO OSNOVNI ELEMENTI PROGRAMIRANJA, KI PREDSTAVLJAJO PODATKE. PYTHON PODPIRA RAZLIČNE TIPE PODATKOV, VKLJUČNO S CELIMI ŠTEVILI, ŠTEVILI S PLAVAJOČO VEJICO, NIZI IN LOGIČNIMI VREDNOSTMI.

TIPI PODATKOV DOLOČAJO, KAKO SE PODATKI LAHKO OBDELUJEJO IN PRIKAZUJEJO V PROGRAMIH.

PRVI PROGRAM:

 lesson_plan_values_and_types.py > ...

```
1  # Examples of different types of values
2  x = 5                                # integer
3  y = 3.14                             # floating-point number
4  name = "Python"                      # string
5  is_fun = True                        # boolean
6
7  print(x + y)                          # Adding two numbers
8  print(name + " is cool!")             # Concatenating text
9  print(is_fun and (x > 2))             # Logical expression
```

2. PRAKTIČNA NALOGA

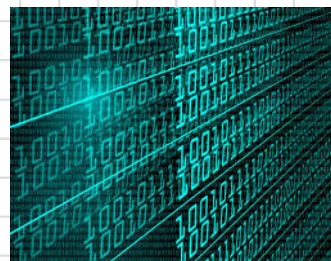
NALOGA: NAPIŠITE PROGRAM V PYTHONU, KI UPORABLJA VREDNOSTI RAZLIČNIH TIPOV. NA PRIMER, SEŠTEJTE DVE ŠTEVILKI, PRIKAŽITE SPOROČILO NA ZASLONU IN UPORABITE LOGIČNI OPERATOR ZA PREVERJANJE POGOJA.

3. DOMAČA NALOGA

NAPIŠITE PROGRAM, KI IZRAČUNA POVPREČJE TREH ŠTEVIL IN NATO PRIKAŽE REZULTAT V SPOROČILU. POSKUSITE EKSPERIMENTIRATI Z RAZLIČNIMI VRSTAMI VREDNOSTI IN OPERATORJEV. SHRANITE PROGRAM IN GA PRINESITE NA NASLEDNJI RAZRED, DA GA PREDSTAVITE SOŠOLCEM.



UČNA PRIPRAVA



PREDMET: PROGRAMIRANJE IN KODIRANJE

TEMA: SPREMENLJIVKE V PROGRAMSKEM JEZIKU PYTHON – LEKCIJA 3

POUDARKI IN CILJI LEKCIJE:

UVOD V KONCEPT SPREMENLJIVK, NJIHOVO VLOGO IN UPORABO PRI PROGRAMIRANJU V JEZIKU PYTHON.

DO KONCA LEKCIJE BI MORALI UČENCI OBVLADATI USTVARJANJE SPREMENLJIVK, DODELJEVANJE VREDNOSTI TER RAZUMEVANJE, KAKO SE SPREMENLJIVKE LAHKO UPORABLJAJO V RAZLIČNIH DELIH PROGRAMA.

POTREBNI MATERIALI:

- RAČUNALNIKI Z NAMEŠČENIM PYTHONOM (NPR. VISUAL STUDIO CODE, PYCHARM ALI DRUGO RAZVOJNO OKOLJE)
- PROJEKTOR ALI ZASLON
- TABELA IN OZNAČEVALCI
- PRILOGE, KI PRIKAZUJEJO KATEGORIJE PODATKOV

UVOD V CHANGELINGE (15 MINUT)

RAZPRAVA (5 MINUT): ZAČNITE Z RAZLAGO, KAJ PREDSTAVLJA SPREMENLJIVKA IN ZAKAJ JE KLJUČNA PRI PROGRAMIRANJU. RAZPRAVLJAJTE O NAČINIH, KAKO LAHKO SPREMENLJIVKE UPORABITE ZA SHRANJEVANJE IN OBDELAVO PODATKOV V PROGRAMIH.

PREDSTAVITEV (10 MINUT): PREDSTAVITE RAZLIČNE TIPE SPREMENLJIVK IN POJASNITE NJIHOVO DELOVANJE. NA PRIMER, PRIKAŽITE, KAKO SHRANITI ŠTEVILKE, BESEDILO ALI LOGIČNE VREDNOSTI V SPREMENLJIVKO. POUDARITE PRAVILA POIMENOVANJA SPREMENLJIVK TER NAJBOLJŠE PRAKSE.

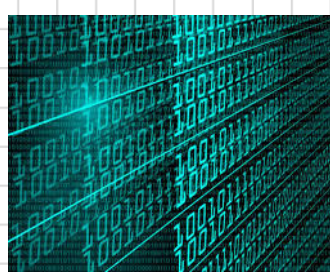
2. UPORABA SPREMENLJIVK V PROGRAMSKEM JEZIKU PYTHON (20 MINUT)

POJASNILO (10 MINUT): PRIKAZUJTE, KAKO USTVARITI IN UPORABLJATI SPREMENLJIVKE V PYTHONU. RAZLOŽITE DODELITEV VREDNOSTI SPREMENLJIVKI, PONOVO VREDNOTENJE SPREMENLJIVKE (NOVO DODELJEVANJE VREDNOSTI) TER PRETVORBO MED RAZLIČNIMI VRSTAMI SPREMENLJIVK (PRETVORBA TIPA).

PRIMER (10 MINUT): NA TABLO ALI ZASLON PROJEKTORJA PRIKAZUJTE NEKAJ PREPROSTIH PRIMEROV KODE, KI ILUSTRIRAJO USTVARJANJE, DODELJEVANJE IN UPORABO SPREMENLJIVK V PROGRAMU. OPIŠITE, KAKO JE MOGOČE SPREMENLJIVKO UPORABITI V VEČ OPERACIJAH TER KAKO SPREMENLJIVKE PRISPEVAJO K VEČJI UČINKOVITOSTI KODE.



UČNA PRIPRAVA



3. PRAKTIČNA DEJAVNOST (20 MINUT)

DEJAVNOST (15 MINUT): UČENCE PROSITE, NAJ NAPIŠEJO PREPROST PROGRAM V PYTHONU, KI UPORABLJA VEČ SPREMENLJIVK. NA PRIMER, USTVARITE NEKAJ SPREMENLJIVK Z RAZLIČNIMI VRSTAMI VREDNOSTI IN NATO IZVEDITE OPERACIJE NA TEH SPREMENLJIVKAH. POMAGAJTE JIM VZPOSTAVITI RAZVOJNO OKOLJE IN POSKRbite, DA LAHKO VSI USPEŠNO IZVAJAJO SVOJ PROGRAM.

DELO V PARIH (5 MINUT): UČENCI V PARIH PREGLEDUJEJO KODO DRUG DRUGEGA. SPODBUJAJTE JIH, DA RAZPRAVLJAJO O MOREBITNIH VPRAŠANJIH ALI TEŽAVAH, KI SE POJAVIJO, TER SI MEDSEBOJNO POMAGAJO PRI IZBOLJŠANJU KODE.

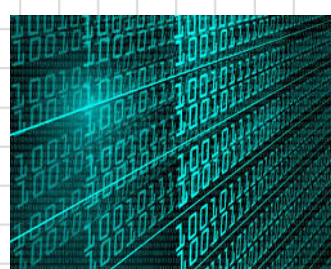
4. VPRAŠANJA IN ODGOVORI TER POVZETEK (5 MINUT)

VPRAŠANJA IN ODGOVORI (3 MINUTE): ODPRITE RAZPRAVO, DA UČENCI LAHKO POSTAVLJAJO VPRAŠANJA. OBRAVNAVAJTE POGOSTE TEŽAVE ALI TOČKE ZMEDE, KI SO SE MORDA POJAVILE MED LEKCIJO. UČENCEM OMOGOČITE, DA IZRAZIJO SVOJE MNENJE O RAZUMEVANJU LEKCIJE.

POVZETEK IN DOMAČA NALOGA (2 MINUTI): POVZEMITE KLJUČNE TOČKE LEKCIJE, OSVETLITE TEMELJNO UPORABO SPREMENLJIVK IN NJIHOV POMEN. KOT DOMAČO NALOGO UČENCE PROSITE, NAJ RAZVIJEJO PROGRAM V PYTHONU, KI VKLJUČUJE RAZLIČNE TIPE SPREMENLJIVK IN IZVAJA RAZLIČNE OPERACIJE NAD NJIMI.



UČNA PRIPRAVA



PREDMET: PROGRAMIRANJE IN KODIRANJE

TEMA: SPREMENLJIVKE V PROGRAMSKEM JEZIKU PYTHON - LEKCIJA 3

SPREMENLJIVKE

GLAVNE TOČKE:

SPREMENLJIVKE SO VSEBNIKI, KI SHRANJUJEJO PODATKE MED IZVAJANJEM PROGRAMA. PRI USTVARJANJU SPREMENLJIVK JIM DODELIMO IME IN PRIPIŠEMO ŽELENO VREDNOST. IMENA SPREMENLJIVK SE DRŽIJO PRAVIL SINTAKSE PYTHONA, KAR VAM OMOGOČA PREPOZNAVANJE IN UPORABO PODATKOV V VAŠEM PROGRAMU.

PRVI PROGRAM:

 variables_example.py > ...

```
1  # Example of using variables
2  x = 10                # integer variable
3  y = 5.5              # floating-point variable
4  name = "Alice"       # string variable
5  is_happy = True      # boolean variable
6
7  # Performing operations with variables
8  sum_result = x + y
9  greeting = "Hello, " + name
10 status = is_happy and (x > y)
11
12 print(sum_result)     # Output the result of the addition
13 print(greeting)      # Output the greeting message
14 print(status)        # Output the logical operation result
```

2. PRAKTIČNA NALOGA

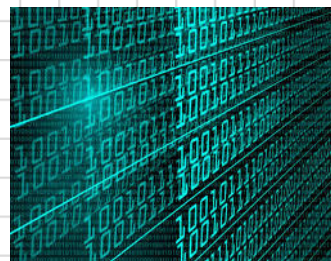
NALOGA: NAPIŠITE PROGRAM V PYTHONU, KI UPORABLJA VREDNOSTI RAZLIČNIH TIPOV. NA PRIMER, SEŠTEJTE DVE ŠTEVILKI, PRIKAŽITE SPOROČILO NA ZASLONU IN UPORABITE LOGIČNI OPERATOR ZA PREVERJANJE POGOJA.

3. DOMAČA NALOGA

NAPIŠITE PROGRAM ZA IZRAČUN POVRŠINE IN OBSEGA KROGA NA PODLAGI DANEGA POLMERA TER UPORABITE SPREMENLJIVKE ZA SHRANJEVANJE VREDNOSTI. SHRANITE PROGRAM IN GA PRINESITE V NASLEDNJI RAZRED, DA GA PREDSTAVITE SOŠOLCEM.



UČNA PRIPRAVA



PREDMET: PROGRAMIRANJE IN KODIRANJE

TEMA: OPERACIJSKI SIGNALI IN OPERANDI V PROGRAMSKEM JEZIKU PYTHON – LEKCIJA 4

POUDARKI IN CILJI LEKCIJE:

UVOD V RAZLIČNE OPERACIJSKE SIGNALE (ARITMETIČNE, PRIMERJALNE, LOGIČNE ITD.) TER NJIHOVO UPORABO V PYTHONU.

DO KONCA LEKCIJE MORAJO UČENCI OBVLADATI USTVARJANJE IN OVREDNOTENJE PREPROSTIH IZRAZOV TER RAZUMETI RAZMERJE MED RAZLIČNIMI OPERACIJAMI IN OPERANDI.

POTREBNI MATERIALI:

- RAČUNALNIKI Z NAMEŠČENIM PYTHONOM (NPR. VISUAL STUDIO CODE, PYCHARM ALI DRUGO RAZVOJNO OKOLJE)
- PROJEKTOR ALI ZASLON
- TABELA IN OZNAČEVALCI
- DODATEK, KI PRIKAŽUJE OPERATIVNE SIGNALE IN OPERANDE.

UVOD V OPERACIJSKE SIGNALE IN OPERANDE (15 MINUT)

RAZPRAVA (5 MINUT): ZAČNITE Z RAZLAGO, KAJ STA ZNAK OPERACIJE IN OPERAND TER KAKO SE UPORABLJATA PRI PROGRAMIRANJU. LAHKO PRIKAŽETE PRIMERE IZ VSAKDANJEGA ŽIVLJENJA, NA PRIMER IZVAJANJE MATEMATIČNIH OPERACIJ.

PREDSTAVITEV (10 MINUT): PRIKAŽUJTE RAZLIČNE OPERACIJSKE SIGNALE (NPR. SEŠTEVANJE, ODŠTEVANJE, MNOŽENJE, DELJENJE ITD.) TER NJIHOVO SINTAKSO V PYTHONU. RAZPRAVLJAJTE O DELOVANJU TEH ZNAKOV S SVOJIMI OPERANDI IN PRIKAŽUJTE RAZLIKE MED ARITMETIČNIMI IN LOGIČNIMI OPERATORJI.

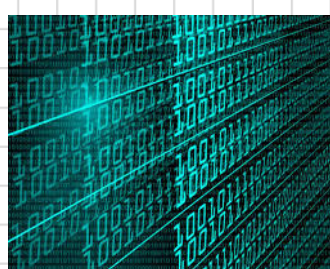
2. UPORABA OPERANDOV IN OPERACIJ V PYTHONU (20 MINUT)

POJASNILO (10 MINUT): PRIKAŽUJTE, KAKO USTVARITI IN UPORABLJATI PREPROSTE IZRAZE V PYTHONU Z UPORABO OPERATORJEV IN OPERANDOV. RAZLOŽITE POSTOPEK VREDNOTENJA IZRAZOV TER NAČIN, KAKO TE OPERACIJE GENERIRAJO REZULTATE.

PRIMER (10 MINUT): NA TABLO ALI ZASLON PROJEKTORJA PRIKAŽUJTE NEKAJ PREPROSTIH PRIMEROV KODE, KI ILUSTRIRAJO UPORABO OSNOVNIH ARITMETIČNIH IN LOGIČNIH OPERATORJEV V PYTHONU. OPIŠITE, KAKO TE OPERACIJE DELUJEJO Z RAZLIČNIMI VRSTAMI OPERANDOV, KOT SO ŠTEVILA IN LOGIČNE VREDNOSTI.



UČNA PRIPRAVA



3. PRAKTIČNA DEJAVNOST (20 MINUT)

DEJAVNOST (15 MINUT): UČENCE PROSITE, NAJ NAPIŠEJO PREPROST PROGRAM V PYTHONU, KI UPORABLJA RAZLIČNE OPERACIJSKE SIMBOLE IN OPERANDE. NA PRIMER, IZRAČUNAJTE OSNOVNE ARITMETIČNE OPERACIJE IN USTVARITE IZRAZE, KI VKLJUČUJEJO LOGIČNE OPERATORJE. POMAGAJTE JIM PRI VZPOSTAVITVI NJIHOVEGA RAZVOJNEGA OKOLJA IN ZAGOTOVITE, DA LAHKO VSI USPEŠNO IZVAJAJO SVOJ PROGRAM.

DELO V PARIH (5 MINUT): UČENCI V PARIH PREGLEDUJEJO KODO DRUG DRUGEGA. SPODBUJAJTE JIH, DA RAZPRAVLJAJO O MOREBITNIH VPRAŠANJIH ALI TEŽAVAH, KI SE POJAVIJO, TER SI MEDSEBOJNO POMAGAJO PRI IZBOLJŠANJU KODE.

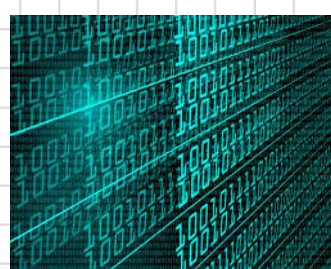
4. VPRAŠANJA IN ODGOVORI TER POVZETEK (5 MINUT)

VPRAŠANJA IN ODGOVORI (3 MINUTE): ODPRITE RAZPRAVO, DA UČENCI LAHKO POSTAVLJAJO VPRAŠANJA. OBRAVNAVAJTE POGOSTE TEŽAVE ALI TOČKE ZMEDE, KI SO SE MORDA POJAVILE MED LEKCIJO. UČENCEM OMOGOČITE, DA IZRAZIJO SVOJE MNENJE O RAZUMEVANJU LEKCIJE.

POVZETEK IN DOMAČA NALOGA (2 MINUTI): POVZEMITE KLJUČNE TOČKE LEKCIJE IN IZPOSTAVITE UPORABO OPERATORJEV TER OPERANDOV V PYTHONU. KOT DOMAČO NALOGO UČENCE SPODBUDITE, NAJ NAPIŠEJO PROGRAM V PYTHONU, KI VKLJUČUJE RAZLIČNE ARITMETIČNE IN LOGIČNE IZRAZE.



UČNA PRIPRAVA



PREDMET: PROGRAMIRANJE IN KODIRANJE

TEMA: OPERACIJSKI SIGNALI IN OPERANDI V PROGRAMSKEM JEZIKU PYTHON - LEKCIJA 4

1. OPERATIVNI SIGNALI IN OPERATIVNI POSTOPKI

GLAVNE TOČKE:

S POMOČJO OPERACIJSKIH SIGNALOV LAHKO IZVAJAMO RAZLIČNE OPERACIJE Z OPERANDI, KI SO LAHKO ŠTEVILKE, BESEDILA ALI LOGIČNE VREDNOSTI.

ARITMETIČNE OPERACIJE (NPR. SEŠTEVANJE, ODŠTEVANJE) IN LOGIČNI OPERATORJI (NPR. IN, ALI, NE) PREDSTAVLJAJO OSNOVNA ORODJA V PROGRAMIRANJU.

POMEMBNO JE RAZUMETI, KAKO TE OPERACIJE VPLIVAJO NA VREDNOST IN VRSTO OPERANDOV.

PRVI PROGRAM:

```
variables_example.py > ...  
1  # Example of using variables  
2  x = 10                # integer variable  
3  y = 5.5              # floating-point variable  
4  name = "Alice"       # string variable  
5  is_happy = True      # boolean variable  
6  
7  # Performing operations with variables  
8  sum_result = x + y  
9  greeting = "Hello, " + name  
10 status = is_happy and (x > y)  
11  
12 print(sum_result)    # Output the result of the addition  
13 print(greeting)      # Output the greeting message  
14 print(status)        # Output the logical operation result
```

2. PRAKTIČNA NALOGA

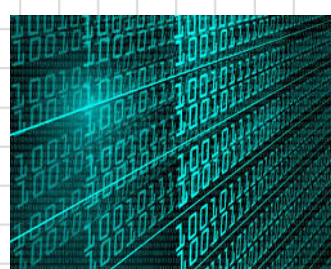
NALOGA: NAPIŠITE PROGRAM V PYTHONU, KI UPORABLJA RAZLIČNE OPERACIJSKE SIGNALE IN OPERANDE. NA PRIMER, IZVEDITE SEŠTEVANJE, DELJENJE IN LOGIČNE OPERACIJE TER NATO OVREDNOTITE IZRAZE.

3. DOMAČA NALOGA

NAPIŠITE PROGRAM, KI IZRAČUNA POVPREČJE DVEH ŠTEVIL IN PRIKAŽE REZULTAT, NATO PA IZVEDE LOGIČNO PREVERJANJE, ALI JE POVPREČJE VEČJE OD DOLOČENE VREDNOSTI. SHRANITE PROGRAM IN GA PRINESITE NA NASLEDNJO URO, DA GA PREDSTAVITE SOŠOLCEM.



UČNA PRIPRAVA



PREDMET: PROGRAMIRANJE IN KODIRANJE

TEMA: POGOJI (IF-ELSE) V PROGRAMSKEM JEZIKU PYTHON – LEKCIJA 5

POUDARKI IN CILJI LEKCIJE:

TA LEKCIJA ŠTUDENTE UVAJA V UPORABO POGOJNIH STAVKOV V PROGRAMSKEM JEZIKU PYTHON, VKLJUČNO S KLJUČNIMI BESEDAMI IF, ELIF IN ELSE.

DO KONCA LEKCIJE MORAJO BITI UČENCI SPOSOBNI UVESTI PREPROSTO POGOJNO LOGIKO V SVOJE PROGRAME TER RAZUMETI POSTOPEK SPREJEMANJA ODLOČITEV V KODI.

POTREBNI MATERIALI:

- RAČUNALNIKI Z NAMEŠČENIM PYTHONOM (NPR. VISUAL STUDIO CODE, PYCHARM ALI DRUGO RAZVOJNO OKOLJE)
- PROJEKTOR ALI ZASLON
- TABELA IN OZNAČEVALCI
- PRILOGA, KI PRIKAŽUJE POGOJNA NAVODILA.

UVOD V POGOJNA NAVODILA (15 MINUT)

RAZPRAVA (5 MINUT): ZAČNITE Z RAZLAGO, ZAKAJ SO POGOJNI STAVKI KLJUČNI PRI PROGRAMIRANJU. RAZPRAVLJAJTE O POMENU ODLOČANJA IN O TEM, KAKO PROGRAMI UPORABLJAJO LOGIČNE IZRAZE ZA DOLOČITEV, KATERA DEJANJA NAJ IZVEDEJO.

PREDSTAVITEV (10 MINUT): PREGLED STRUKTURE POGOJNIH STAVKOV V PYTHONU. OPIŠITE VLOGO KLJUČNIH BESED IF, ELIF IN ELSE TER PRIKAŽITE PRIMERE IZ VSAKDANJEGA ŽIVLJENJA, KJER JE POTREBNO SPREJEMATI ODLOČITVE, NATO PA PREDSTAVITE PRIMER KODE V PYTHONU.

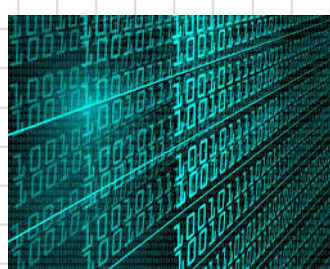
2. UPORABA POGOJNIH STAVKOV V PYTHONU (20 MINUT)

POJASNILO (10 MINUT): PRIKAŽUJTE, KAKO JE MOGOČE UPORABITI POGOJNE STAVKE V PYTHONU. RAZLOŽITE DELOVANJE STRUKTURE IF-ELSE TER UPORABO LOGIČNIH IZRAZOV ZA PREVERJANJE POGOJEV. NAVEDITE PRIMERE RAZLIČNIH SITUACIJ, KOT SO DOLOČANJE URE NA PODLAGI TRENUTNEGA ČASA ALI ODLOČANJE O TEM, ALI JE NEKDO OPRAVIL IZPIT NA PODLAGI DOSEŽENEGA REZULTATA.

PRIMER (10 MINUT): NA TABLO ALI ZASLON PROJEKTORJA ZAPIŠITE PREPROSTO PRIMERNO KODO, KI PONAŽARJA UPORABO STRUKTURE IF-ELSE. NA PRIMER, PROGRAM, KI ODLOČA, ALI JE OSEBA UPRAVIČENA DO POPUSTA ALI PA JE ŠTEVILO POZITIVNO, NEGATIVNO ALI NIČ.



UČNA PRIPRAVA



3. PRAKTIČNA DEJAVNOST (20 MINUT)

DEJAVNOST (15 MINUT): UČENCE PROSITE, NAJ NAPIŠEJO PREPROST PROGRAM V PYTHONU, KI VKLJUČUJE STRUKTURO IF-ELSE. NA PRIMER, NAJ NAPIŠEJO PROGRAM, KI PREVERI STAROST UPORABNIKA IN SE ODLOČI, ALI LAHKO UPORABNIK VSTOPI NA DOLOČEN DOGODEK. POMAGAJTE JIM PRI VZPOSTAVITVI NJIHOVEGA RAZVOJNEGA OKOLJA IN POSKRbite, DA LAHKO VSI USPEŠNO IZVAJAJO SVOJ PROGRAM.

DELO V PARIH (5 MINUT): UČENCI V PARIH PREGLEDUJEJO KODO DRUG DRUGEGA. SPODBUJAJTE JIH, DA RAZPRAVLJAJO O MOREBITNIH VPRAŠANJIH ALI TEŽAVAH, KI SE POJAVIJO, TER SI MEDSEBOJNO POMAGAJO PRI IZBOLJŠANJU KODE.

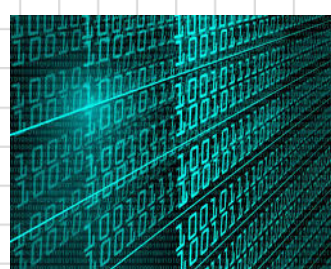
VPRAŠANJA IN ODGOVORI TER POVZETEK (5 MINUT)

VPRAŠANJA IN ODGOVORI (3 MINUTE): ODprite RAZPRAVO, DA UČENCI LAHKO POSTAVLJAJO VPRAŠANJA. OBRAVNAVAJTE POGOSTE TEŽAVE ALI TOČKE ZMEDE, KI SO SE MORDA POJAVILE MED LEKCIJO. UČENCEM OMOGOČITE, DA IZRAZIJO SVOJE MNENJE O RAZUMEVANJU LEKCIJE.

POVZETEK IN DOMAČA NALOGA (2 MINUTI): POVZEMITE KLJUČNE TOČKE LEKCIJE IN IZPOSTAVITE POMEN POGOJNIH IZJAV. KOT DOMAČO NALOGO UČENCE PROSITE, NAJ RAZVIJEJO PROGRAM V PYTHONU, KI MODELIRA PREPROST POSTOPEK ODLOČANJA S STRUKTURO IF-ELSE.



UČNA PRIPRAVA



PREDMET: PROGRAMIRANJE IN KODIRANJE

TEMA: POGOJI (IF-ELSE) V PROGRAMSKEM JEZIKU PYTHON - LEKCIJA 5

POGOJNA NAVODILA

GLAVNE TOČKE:

POGOJNE STAVKE OMOGOČAJO PROGRAMU, DA SPREJEMA ODLOČITVE GLEDE NA IZPOLNITEV DOLOČENIH POGOJEV.

KLJUČNE BESEDE IF, ELIF IN ELSE SE UPORABLJAJO ZA OBRAVNAVO RAZLIČNIH SITUACIJ V VAŠI KODI.

LOGIČNI IZRAZI DOLOČAJO, KATERI SEGMENT KODE SE IZVEDE.

PRVI PROGRAM:

 if_else_examples.py > ...

```
1  # Example of using if-else statements to determine the time of day
2  hour = int(input("Enter the current hour (0-23): "))
3
4  if 5 <= hour < 12:
5      |   print("Good morning!")
6  elif 12 <= hour < 18:
7      |   print("Good afternoon!")
8  elif 18 <= hour < 22:
9      |   print("Good evening!")
10 else:
11     |   print("Good night!")
```

2. PRAKTIČNA NALOGA

NALOGA: NAPIŠITE PROGRAM V PYTHONU, KI UPORABLJA STRUKTURO IF-ELSE ZA PREVERJANJE RAZLIČNIH POGOJEV. ODLOČITE SE, NA PRIMER, ALI JE ŠTEVILO SODO ALI LIHO, ALI PREVERITE, ALI JE OSEBA UPRAVIČENA DO POPUSTA GLEDE NA STAROST.

3. DOMAČA NALOGA

NAPIŠITE PROGRAM, KI PREVERI TEMPERATURO IN SE ODLOČI, ALI JE VREME HLADNO, TOPLO ALI VROČE. UPORABITE STRUKTURO IF-ELSE ZA ODLOČANJE IN SHRANITE PROGRAM ZA PREDSTAVITEV V NASLEDNJEM RAZREDU.



CNC

01. 01. 2024 –31. 12. 2025



**Funded by
the European Union**



**Spojená škola
Kollárova 17, Sečovce**

**Móricz Zsigmond
Református Kollégium,
Gimnázium, Technikum,
Általános Iskola és Óvoda**



**SOLSKI CENTER
SLOVENJ GRADEC**

2023-2-SK01-KA210-VET-000184496