

МОДЕЛ ПРИПРЕМЕ ЧАСА

Информатика и рачунарство за шести разред разред

Наставна тема:	РАЧУНАРСТВО
Редни број часа:	4.
Циљ часа:	<ul style="list-style-type: none">• Упознавање ученика са концептом корњача графике.• Упознавање ученика са библиотеком turtle.• Упознавање ученика са основним наредбама библиотеке turtle.• Упознавање ученика са начинима исцртавања неких елементарних фигура применом корњача графике.
Исход часа:	Ученик ће након часа бити у стању да: <ul style="list-style-type: none">• уме да укључи библиотеку turtle• користи једноставније наредбе библиотеке turtle• самостално црта једноставније геометријске фигуре• користи понављање на домену корњача графике за исцртавање једноставнијих геометријских фигура
Облици рада:	Фронтални, индивидуални
Наставне методе:	Метода усменог излагања, илустративно – демонстративна метода, метода вођеног разговора
Место реализације:	Рачунарски кабинет

Активности наставника:

КОРЊАЧА ГРАФИКА

На почетку часа наставник би требало укратко да упозна ученике са корњача графиком¹.

Корњача графика је леп начин да се ученицима представе основни концепти програмирања. Концепт корњача графике је такав да се неки замишљени лик креће по белој подлози и за собом оставља траг који представља цртеж. На почетку (у Logo језику), лик који је остављао траг била је корњача, тако да одатле потиче назив *корњача графика*.

Може се ученицима демонстрирати неки од примера који се налазе на сајту code.org. Наставник би требало са ученицима да продискутује да ли су некада раније видели (радили) овакве задатке.

Корњача графика се, за разлику од Карела, може радити независно од Петљиног портала, довољно је да је на рачунару инсталиран Python.

Како би ученици могли да цртају фигуре у корњача графици, потребно је скренути им пажњу да је потребно прво да укључите библиотеку turtle, као и да науче одређене наредбе.

БИБЛИОТЕКА TURTLE И НАРЕДБЕ КОРЊАЧА ГРАФИКЕ

Пре писања програма у корњача графици потребно је укључити библиотеку Python-а и то се ради на следећи начин:

```
import turtle
```

1 https://en.wikipedia.org/wiki/Turtle_graphics

Скренути пажњу ученицима да је библиотеку turtle потребно укључити непосредно пре коришћења наредби ове библиотеке. То обично чинимо навођењем претходног реда на самом почетку програма.

Наредбе позивамо тако што након `turtle` наведемо неку од наредби (`turtle.forward()`). Како бисмо уштедели време приликом писања програма, можемо на самом почетку приликом укључивања библиотеке turtle, да наведемо ред

```
import turtle as t
```

и након овога, све наредбе умсето са `turtle.naredba` позивамо на следећи начин `t.forward()`.

НАРЕДБЕ:

1.	Показати како се црта линија дужине 100 пиксела. <code>turtle.forward(n)</code> <i>корњача се помера напред за n корака</i>	Задати ученицима да сами нацртају линију дужине 80 пиксела.
2.	Показати како се мења боја линије. <code>turtle.color(c)</code> <i>боја оловке се поставља на боју c (назив је на енглеском језику, нпр. "red", "green", "blue")</i>	Задати ученицима да замене боју линије коју су нацртали у неку од боја.
3.	Показати како се мења дебљина линије. <code>turtle.width(n)</code> <i>дебљина оловке се поставља на вредност n</i>	Задати ученицима да промене дебљину своје линије.
4.	Показати како се може нацртати испрекидана линија. <code>turtle.penup()</code> <i>корњача подиже оловку</i> <code>turtle.pendown()</code> <i>корњача спушта оловку</i>	Задати ученицима да нацртају испрекидану линију.
5.	Показати ученицима како могу да промене изглед лика који оставља траг. <code>turtle.shape(s)</code> <i>корњача мења свој облик (нпр. "arrow", "turtle", "circle")</i>	Задати ученицима да промене изглед лика на круг.

6.	Показати како се постиже ефекат отисака. <code>turtle.stamp()</code> <i>корњача оставља свој отисак</i>	Задати ученицима да направе ефекат да корњача оставља отисак.
7.	Показати како се може променити брзина исцртавања. <code>turtle.speed(n)</code> <i>корњача мења своју брзину кретања од 0 (најбрже) до 10 (најспорије)</i>	Задати ученицима да промене брзину исцртавања трага.
8.	Показати како се може окретати лик који исцртава за одређен број степени у одређеном правцу. <code>turtle.left(n)</code> <i>корњача се окреће налево за n степени</i> <code>turtle.right(n)</code> <i>корњача се окреће надесно за n степени</i>	Задати ученицима да ово самостално пробају.

ЛИНИЈСКИ ПРОГРАМИ И ПОНАВЉАЊЕ

Након увођења наредби са ученицима би требало урадити неке од примера са портала [petlja.org](https://www.petlja.org). Леп пример на коме се може на самом почетку видети линијска структура програма, али и понављање је пример *Квадрат* на адреси <https://www.petlja.org/biblioteka/r/lekcije/prirucnik-python/kornjaca>

Ученицима би требало скренути пажњу да уколико у програму постоји наредба (или више наредби које се понављају одређен број пута), програмски језици нам дају наредбе којима можемо лако организовати понављање.

Погледајмо наведени пример:

```
import turtle
turtle.forward(100) # иди напред 100 корака
turtle.left(90)    # окрени се 90 степени налево
turtle.forward(100) # иди напред 100 корак
turtle.left(90)    # окрени се 90 степени налево
turtle.forward(100) # иди напред 100 корака
turtle.left(90)    # окрени се 90 степени налево
turtle.forward(100) # иди напред 100 корака
turtle.left(90)    # окрени се 90 степени налево
```

Уочимо да се две наредбе

```
turtle.forward(100)
turtle.left(90)
```

повнављају четири пута и управо из овог разлога користимо петљу `for`.

Ученике подсетити на синтаксу петље `for`, скренути пажњу на значај правилног назубљивања кода, на наредбе које су и визуелно издвојене у телу петље и да се управо оне понављају одређен број пута.

Приказати ученицима петљу for:

```
for i in range(n):  
    naredba1()  
    naredba2()
```

и још једном поновити које се наредбе понављају и колико пута.

Након овога, рад на овом примеру препустити ученицима и дати им инструкцију да сами организују код тако да се користи понављање.

ВЕЖБАЊЕ

Заједно са ученицима анализирати пример *Квадратни сигнал* и када се идентификује низ једноставних облика који се понављају задати ученицима да самостално напишу програм.

Активности ученика:

КОРЊАЧА ГРАФИКА

1. Ученик пажљиво слуша наставниково излагање у вези са корњача графиком
2. Ученик учествује у разговору на тему корњача графика и задаци са сајта code.org
3. Упутити ученике на бидео лекцију



БИБЛИОТЕКА TURTLE И НАРЕДБЕ КОРЊАЧА ГРАФИКЕ

1. Ученик прати наставниково објашњење како се користи библиотека turtle
2. Ученик пажљиво прати док наставник уводи једну за другом наредбу из библиотеке turtle
3. Ученик на свом рачунару пише кратке кодове којима усваја наредбе библиотеке turtle

ЛИНИЈСКИ ПРОГРАМИ И ПОНАВЉАЊЕ

1. Ученик пажљиво прати излагање наставника који анализира опис исцртавања квадрата коришћењем линијске структуре
2. Ученик се труди да уочи низ наредби које се понављају
3. Ученик се труди да одговори на питање колико се пута уочени низ наредби понавља
4. Ученик сарађује са наставником и осталим ученицима у дискусији око тога који тип

понављања овде користимо.

5. Ученик уме да образложи свој одговор.
6. Ученик пажљиво слуша наставниково образложење датог одговора.

ВЕЖБАЊЕ

1. Ученик пажљиво прати излагање наставника који анализира опис исцртавања квадратног сигнала
2. Ученик се труди да уочи низ наредби које се понављају
3. Ученик се труди да одговори на питање колико се пута уочени низ наредби понавља
4. Ученик сарађује са наставником и осталим ученицима у дискусији око тога који тип понављања овде користимо.
5. Ученик уме да образложи свој одговор.
6. Ученик самостално пише програм који је решење овог задатка.
7. Упутити ученика на видео лекцију:



ДОМАЋИ ЗАДАТАК:

На адреси <https://www.petlja.org/biblioteka/r/lekcije/prirucnik-python/kornjaca> задати ученицима за домаћи пример:

Испрекидана линија

Слободно цртање

Упутити ученике на видео лекцију о понављању које ће се детаљније обрадити на наредном часу.

Kornjača grafika - Ponavljanje (petlje)

Pogledaj lekciju



Препоручени материјал:

1. <https://youtu.be/q0btftOGHyE>
2. <https://youtu.be/Wiq2DawUT9c>
3. <https://youtu.be/vq9H7E2WDt4>