Южный федеральный университет
Институт математики, механики и компьютерных наук им. И. И. Воровича
направление подготовки «Прикладная математика и информатика»

Работа с модулем turtle[1]

Библиотека turtle позволяет рисовать на экране простые рисунки. Рисование осуществляется путем перемещения по экрану черепашки, управление которой осуществляется при помощи инструкций Python.
Программу, использующую модуль turtle надо запускать в графической оболочке. Все программы, работающие с черепашкой, должны начинаться с команд `import turtle` и `turtle.reset()` , а заканчиваться строкой `turtle.mainloop()` или `done()` (оставить графическое окно на экране после выполнения программы) или `turtle.exitonclick()` (закрытие графического окна по щелчку мыши).

Основные команды черепашки:

команда	**действие черепашки**
forward(X) | fd()	пройти вперёд на X пикселей
backward(X) | bk() | back()	пройти назад на X пикселей
left(X) | lt()	повернуть налево на X градусов
right(X) | rt()	повернуть направо на X градусов
circle(r)	нарисовать окружность радиуса $\|r\|$, центр которой находится слева от черепашки, если $r < 0$ и справа, если $r > 0$
circle(r, a)	нарисовать дугу радиуса $\|r\|$ и градусной мерой a. Дуга рисуется против часовой стрелки, если $r > 0$ и по часовой стрелке, если $r < 0$
penup() | pu() | up()	не оставлять след по пути движения (по умолчанию включено)
pendown() | pd() | down()	оставлять след по пути движения
shape(X)	изменить значок черепашки. Возможные значения: 'arrow', 'turtle', 'circle', 'square', 'triangle', 'classic'
reset()	возвращение черепашки в начальное положение
stamp()	нарисовать копию черепахи в текущем месте
color()	установить цвет
begin_fill()	вызывыается перед рисованием фигуры, которую нужно закрасить
end_fill()	вызывается после окончания рисования фигуры
width() | pensize()	толщина линии
speed(X)	изменение скорости движения черепахи. X может быть строкой или числом от 0.5 до 10. Возможные варианты: 'fastest' или 0, 'fast' или 10, 'normal' или 6, 'slow' или 3, 'slowest' или 1
goto(x, y)	переместить черепашку в точку (x, y)
clear()	очистка экрана
write(text)	вывести `text` на экран в точке нахождения черепашки

Со списком всех команд можно ознакомиться на docs.python.org

[1]Разработано А.М. Филимоновой (кафедра ВМиМФ мехмата ЮФУ)

Пример 1. Рисование геометрических фигур

```
from turtle import *
t=Turtle()
t.screen.bgcolor("black") # смена цвета фона
t.color("red")            # смена цвета черепашки
t.hideturtle()            # черепашка невидима

def change_pos(x,y):      # функция, двигающая черепашку на угол x и расстояние y
    t.up()
    t.right(x)
    t.forward(y)
    t.right(x)
    t.pendown()

def square(length):       # функция рисования квадрата
    for steps in range(4):
        t.fd(length)
        t.left(90)

def slanted_rectangle(length,width,angle):
    t.setheading(angle)   # функция рисования прямоугольника
    for steps in range(2):
        t.fd(width)
        t.left(90)
        t.fd(length)
        t.left(90)

def triangle(length,angle=120):
    for steps in range(3):# функция рисования треугольника
        t.fd(length)
        t.left(angle)

def star():               # функция рисования звезды
    for i in range(5):
        t.forward(150)
        t.right(144)

slanted_rectangle(length=200,angle=45,width=100)

change_pos(90, 300)
t.color("blue")
square(100)

change_pos(90, 450)
t.color("green")
t.begin_fill()            # нарисуем закрашенную фигуру
triangle(120)
t.end_fill()

change_pos(-180, 100)
t.color("white")
t.circle(50,180)          # нарисуем полукруг
```

```
change_pos(90, -500)
t.color("purple")
t.write("Turtle is cool! ",move=True,align="center",font=("Freestyle Script",50,"normal"))

change_pos(500, -400)
star()
mainloop()
```

Пример 2. Случайное движение черепашки

```
import random
from turtle import *
t=Turtle()
t.screen.bgcolor("black")

def random_drawing(turns,distance):
    for x in range(turns):
        right=t.right(random.randint(0,360))
        left=t.left(random.randint(0,360))
        t.color(random.choice(["blue","red","green", "purple", "white"]))
        random.choice([right,left])
        t.fd(distance)

random_drawing(100,50)

exitonclick()
```

Пример 3. Рисование фракталов

```
from turtle import *

def tree(sz, level, angle):

    if level > 0:
        colormode(255)
        pencolor(0, 255//level, 0) # установим какой-то оттенок зелёного
        fd(sz) # движение вперед на sz пикселей
        rt(angle) #
        tree(0.8 * sz, level-1, angle) # отрисовка правого поддерева

        pencolor(0, 255//level, 0)
        lt( 2 * angle ) # поворот налево на angle градусов
        tree(0.8 * sz, level-1, angle) # отрисовка левого поддерева
        pencolor(0, 255//level, 0)
        rt(angle) # поворот направо на angle градусов
        fd(-sz) # движение вперед на -sz пикселей (т.е. назад)


speed("fastest") # установка скорости движения черепашки
rt(-90) # поворот на 90 градусов влево (направление движения вверх)
angle = 50 # угол отклонения при отрисовке
size = 80 # начальная длина одной "ветки"
rec_level = 7 # глубина рекурсии

tree(size, rec_level, angle)
```

Задачи для самостоятельного решения.

---

1. Нарисуйте 10 вложенных друг в друга квадратов. Рисование одного квадрата оформить в виде функции.
2. Нарисуйте квадратную спираль.
3. Доработайте **Пример 1** так, чтобы случайная фигура появлялась в случайном месте
4. Используя **Пример 4**, нарисуйте разноцветный лес.
5. Нарисуйте что хотите :)