

УДК 372.862

*Ахмедова Ирода Нурмухамедовна,
старший преподаватель,
Ташкентский международный университет образования,
Республика Узбекистан, город Ташкент*

ПРИМЕНЕНИЕ МОДУЛЯ TURTLE В РЕШЕНИИ МАТЕМАТИЧЕСКИХ ЗАДАЧ НА PYTHON

Аннотация: В данной статье, рассмотрены вопросы изучения интеграции методов Python с возможностями модуля Turtle в рисовании фигур и работа с текстовыми файлами.

Ключевые слова: геометрические фигуры, методы Python, модуль Turtle, создание текстового файла, фрактальная графика, вычисление площади прямоугольника.

*Akhmedova Iroda Nurmukhamedovna,
Senior Lecturer,
Tashkent International University of Education
Republic of Uzbekistan, Tashkent*

USING THE TURTLE MODULE TO SOLVE MATH PROBLEMS IN PYTHON

Abstract: In this article, the issues of studying the integration of Python methods with the capabilities of the Turtle module in drawing shapes and working with text files are considered.

Key words: geometric shapes, Python methods, Turtle module, creating a text file, fractal graphics, calculating the area of a rectangle.

В Стратегических целях развития системы высшего образования предусмотрено: повышение качества подготовки высококвалифицированных кадров, развитие человеческого капитала на основе требований рынка труда для модернизации и стабильного социально-экономического развития

страны; создание необходимых условий для повышения уровня охвата высшим образованием, подготовки высококвалифицированных, креативно и системно мыслящих кадров на основе международных стандартов, способных самостоятельно принимать решения для реализации их интеллектуальных способностей и формирования в качестве духовно развитой личности [1]. В образовательной системе изучение среды программирования создает фантастическое количество логических мыслей и задач начиная от простого до всячески изощренных. Типы задачи, для которых нужно и необходимо представление разнородной графики, использует фрактальную графику. Фрактальная графика – графика, созданная программным кодом. В этом аспекте можно отметить широкие спектры Python - это универсальный, динамический, высокоуровневый интерпретируемый язык программирования. Модуль Turtle языка Python предоставляет примитивы черепашьей графики, как объектно-ориентированными, так и процедурно-ориентированными способами. Поскольку он использует Tkinter для базовой графики, ему необходима версия Python3, установленная с поддержкой Tk [2].

В цель работы поставлена - разработать программу, способную реализовать создание геометрических фигур на основе вычисляемых параметров, для которой назначены следующие задачи:-

- ✓ создать *.txt – файл для хранения данных, в возможности их передачи по сетям, на носителе и представления на иных приложениях.
- ✓ считать из файла данные, по которым вычислить площади фигур n-ного количества. Результаты сохранить в файл.
- ✓ по результатам нарисовать фигуры.

Актуальность данной работы составляет расширение базы знаний работы с библиотеками, методами, операторами программного языка Python, умение управлять данными в текстовом файле, а также переносить их на графику.

Расположение фигур на плоскости, созданных фрактальной графикой, могут дать объяснение в разных толкованиях: размещение их на плоскости и относительные размеры, определение общей площади и периметров, разработка дизайна и т.п. Каждое из них находят свое определение в разных сферах и отраслях.

В постановке дано значение n целого типа *int*, который определяет количество, и n -е количество пар чисел a и b вещественного типа *float*, определяющие длины сторон каждого прямоугольника. Значения нужно вводить с клавиатуры, а также сохранить в файле *faul.txt*, это особенно важно, если их количество превышает 10^2 . Данные форматировано считываются в лист, производится вычисление площади фигур. Полученные результаты нужно сохранить в текстовом файле *Square_REC.txt* и нарисовать фигуры с помощью модуля *turtle*.

Программа реализовано в среде *IDLE Shell 3.9.7. (Python 3.9 64-bit)*, среда поддерживает разных встроенных модулей, в том числе *tkinter* и *turtle*. Ниже перечислены процессы (операции, функции, расчеты), которые введены в программу:

Для печати системной и форматированной даты и времени были использованы методы *date.today()*, *time.asctime*, *time.localtime*, *time.time()* модуля *DataTime*.

Для записи данных и результатов в текстовые файлы использован метод *F.writelines()*. Метод файлового объекта *close()* автоматически закрывает файл, при этом теряется любая несохраненная информация. Работать с файлом (читать, записывать) после этого нельзя.

Метод *lines.split(' ')* разбивает текстовую строку на множество текстовых строк по признаку символа, указанного в скобке.

Операция *num = float(s)* конвертирует строку в вещественное число и метод *T2.append(num)* добавляет значения в виде массива чисел в структуру данных *list T2*.

Вычисление площади прямоугольников:

Цикл пока i меньше длины $T2$

`while i <= len(T2):`

`a=T2[i-1]`

`b=T2[i]`

`S=abs(a*b)`

Вывод результатов с функцией:

`print ("Площадь прямоугольника", T2[i-1], 'x', T2[i], '=', "{:6.4f}".format(S))`

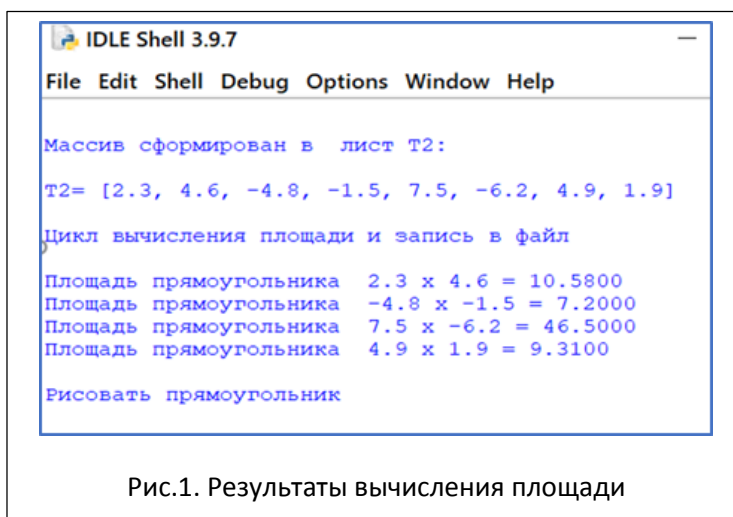


Рис.1. Результаты вычисления площади

Для рисования прямоугольников в модуле Turtle использованы:

`turtle.left(90)` - поворот на 90 градусов против часовой стрелки

`turtle.forward(50)` - перемещение на 50 точек вперед

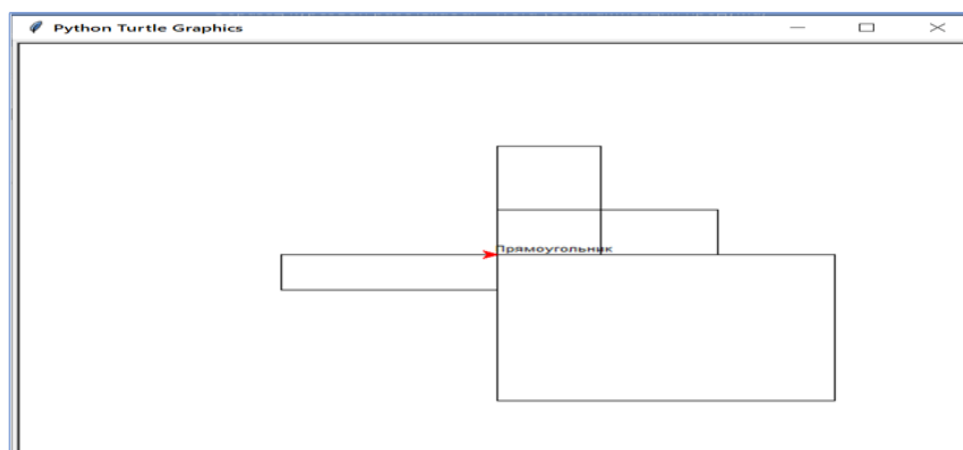


Рис.2. Фигуры прямоугольника, нарисованные модулем Turtle

На рисунке 2 показано, что прямоугольники, разложенные на разные стороны плоскости с началом в общей точке, имеют общую площадь. Из которого следует, что выполненная работа имеет актуальную значимость в

решении типовых задач в разной интерпретации, также с другими фигурами задача будет иметь иную интонацию, на пример, размещение трехмерных фигур в друг друга. Такое толкование может быть использовано для задач технического, земельного, архитектурного типа и др.

В заключении, можно сказать о том, что в процессе работы были изучены разные конструкции алгоритмизации, возможности языка программирования Python, насколько он многоаспектен, а также модуля Turtle с применением примитивов дают базовые понятия в изучение создания фрактальной графики и ее возможностей.

Развитые и гибкие интеграционные возможности Python являются его основным преимуществом в качестве языка для интеграции приложений.

Использованные источники:

1. Ахмедова И.Н. Целевая интенсивная подготовка - форма интеграционного процесса науки, образования и производство. [Электронный ресурс]// Экономика и социум.-2020.- №6(73) URL: https://www.iupr.ru/_files/ugd/b06fdc_86d065d69421409aa4e49e82c9c59fa3.pdf?index=true (дата обращения: 15.06.2023).
2. Turtle — Черепашня графика. URL: <file:///E:/PYTHON/TURTLE%20grafik/turtle%20%E2%80%94%D0%A7%D0%B5%D1%80%D0%B5%D0%BF%D0%B0%D1%88%D1%8C%D1%8F%D0%B3%D1%80%D0%B0%D1%84%D0%B8%D0%BA%D0%B0%20%20Python%203.html> (дата обращения 15.06.2023г.).
3. Саммерфилд М. Программирование на Python 3. Подробное руководство. — М.: Символ-Плюс, 2009. — 608 с.: ISBN: 978-5-93286-161-5.
4. Swaroop С Н (Перевод: Владимир Смоляр). A Byte of Python (Russian), Версия 2.02. URL:<http://wombat.org.ua/AByteOfPython/AByteofPythonRussian-2.02.pdf> (дата обращения 11.01.2023г.)