

DOI: 10.32517/2221-1993-2025-24-2-59-64

**Л. М. Коренюгина**

Сибирский федеральный университет, г. Красноярск, Россия;

Средняя школа № 63, г. Красноярск, Россия

МЕТОДЫ ФОРМИРОВАНИЯ ВИЗУАЛЬНОГО ЦИФРОВОГО ОБРАЗОВАТЕЛЬНОГО КОНТЕНТА С ПОМОЩЬЮ ЯЗЫКА PYTHON*

Аннотация

Цифровой образовательный контент — это любой информационный материал, созданный в образовательных целях, который может быть разработан и распространен в цифровом формате. Цель данной статьи — продемонстрировать (на примере языка программирования Python и использования возможностей его графических модулей и библиотек), как можно расширить круг инструментов для создания цифрового образовательного контента. В статье приведены примеры использования языка Python для создания иллюстративных материалов к решению задач из курсов информатики и математики. Практическая значимость исследования заключается в демонстрации возможности применения программирования для создания визуального образовательного контента, который может использоваться учителями для разработки различных типов самостоятельных работ, создания иллюстративного материала для презентаций, тестов и решения других образовательных задач. Также предложенная методика может быть использована для формирования комплекса заданий для самостоятельной работы учащихся при подготовке презентаций, тематических обзоров и т. д. Создание визуальных моделей средствами языка программирования развивает творческие способности учащихся, формирует устойчивые навыки владения языком программирования и применения его для решения широкого круга задач.

Ключевые слова: цифровой образовательный контент, визуализация, цифровая образовательная среда, язык программирования Python, turtle, matplotlib.

Информация об авторе

Коренюгина Людмила Михайловна, старший преподаватель кафедры систем искусственного интеллекта, Институт космических и информационных технологий, Сибирский федеральный университет, г. Красноярск, Россия; учитель информатики, средняя школа № 63, г. Красноярск, Россия; *e-mail*: korenugina71@yandex.ru

1. Введение

Цифровой образовательный контент — это любой информационный материал (текст, изображения, видео-, аудио-, мультимедийные материалы), созданный

в образовательных целях, который может быть разработан и распространен в цифровом формате. *Процесс отображения обучающих материалов в графической или визуальной форме называется визуализацией цифрового образовательного контента.* Она может

* Материалы к статье можно скачать на сайте ИНФО: http://infojournal.ru/journals/school/school_02-2025/

L. M. Korenyugina

Siberian Federal University, Krasnoyarsk, Russia;

School 63, Krasnoyarsk, Russia

METHODS FOR CREATING VISUAL DIGITAL EDUCATIONAL CONTENT USING PYTHON LANGUAGE

Abstract

Digital educational content is any information material created for educational purposes that can be developed and distributed in digital format. The purpose of the article is to demonstrate (using the Python programming language and its graphical modules and libraries as an example) how to expand the range of tools for creating digital educational content. The article provides examples of using Python to create illustrative materials for solving tasks in informatics and mathematics courses. The practical significance of the study lies in demonstrating the possibility of using programming to create visual educational content that can be used by teachers to develop various types of independent work, create illustrative material for presentations, tests and solve other educational problems. The proposed methodology can also be used to form a set of tasks for independent work of students when preparing presentations, thematic reviews, etc. Creating visual models using a programming language develops students' creativity, forms sustainable skills in proficiency in a programming language and its application to solving a wide range of problems.

Keywords: digital educational content, visualization, digital educational environment, Python programming language, turtle, matplotlib.

Information about the author

Lyudmila M. Korenyugina, Senior Lecturer at the Department of Artificial Intelligence Systems, Institute of Space and Information Technologies, Siberian Federal University, Krasnoyarsk, Russia; Informatics Teacher, School 63, Krasnoyarsk, Russia; *e-mail*: korenugina71@yandex.ru

```

import turtle
t=turtle.Turtle()

t.pensize(2)
t.goto(-200,0)
t.goto(250,0)
t.write('x',font=('arial',24))
t.goto(0,0)
t.goto(0,250)
t.write('y',font=('arial',24))

t.goto(0,-200)
t.goto(0,0)
t.write('(0,0)',font=('arial',15))
t.goto(250,0)

t=turtle.Turtle()

t.up()
t.goto(0,200)
t.write('(0,200)',font=('arial',15))
t.up()
t.goto(60,120)
t.write('(60,120)',font=('arial',15))
t.up()
t.goto(80,40)
t.write('(80,40)',font=('arial',15))
t.up()
t.goto(100,-60)
t.write('(100,-60)',font=('arial',15))
t.up()
t.goto(-180,-60)
t.write('(-100,-60)',font=('arial',15))
t.up()
t.goto(-160,40)
t.write('(-80,40)',font=('arial',15))
t.up()
t.goto(-150,120)
t.write('(-60,120)',font=('arial',15))
t.up()

t.left(90)
t.goto(0,250)

t=turtle.Turtle()
t.up()
t.goto(200,200)
for i in range(22):
    t.up()
    t.goto(-200+i*20,250)
    t.down()
    t.goto(-200+i*20,-200)
for i in range(20):
    t.up()
    t.goto(-200,240-i*20)
    t.down()
    t.goto(240,240-i*20)
t.pensize(4)

t.pencolor('green')
t.up()
t.goto(0,200)
t.down()

t.goto(60,120)
t.goto(20,120)

t.goto(80,40)
t.goto(20,40)

t.goto(100,-60)
t.goto(-100,-60)

t.goto(-20,40)

t.goto(-80,40)
t.goto(-20,120)
t.goto(-60,120)
t.goto(0,200)
t.screen.exitonclick()
t.screen.mainloop()

```

Листинг 1. Решение примера 1

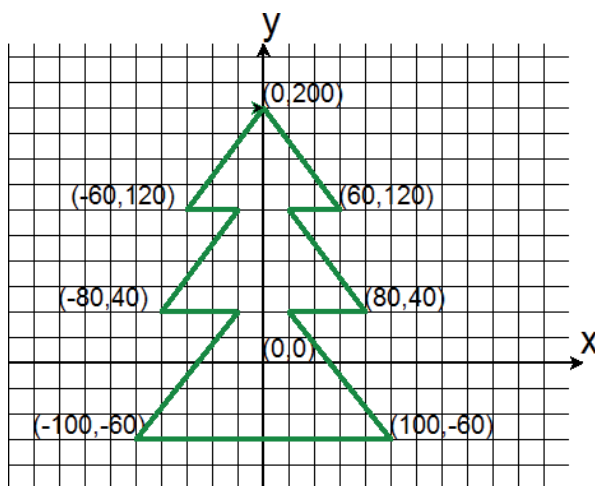


Рис. 1. Результат работы программы примера 1

использоваться для представления диаграмм, графиков, схем и других типов информации. Согласно множеству научных исследований, человек получает 90 % информации из окружающей среды с помощью зрения, поэтому использование в образовании визуального образовательного контента является эффективным способом передачи информации [7, 9].

При создании образовательного контента возникает необходимость в качественном и привлекательном визуальном оформлении учебных материалов [1]. Учителя часто сталкиваются с необходимостью оформить визуально привлекательно и аккуратно материалы для презентаций и печатных работ, используемых в учебном процессе. В результате они используют доступные им инструменты графического редактора (например, Paint) или графические возможности текстового редактора (например, Word). Многие учителя используют готовые материалы и разработки к урокам, размещенные в сети Интернет.

Мы предлагаем шире посмотреть на разработку учебных материалов и использовать возможности языка Python в качестве еще одного инструмента для создания графических объектов и пополнения коллекции материалов для обучения.

В данной статье рассматриваются примеры использования при оформлении учебных материалов модуля *turtle* [4] и библиотеки *matplotlib* языка программирования Python.

2. Методы создания графических учебных материалов

В языке Python есть несколько графических модулей, которые можно использовать для создания графических материалов по информатике, математике и другим дисциплинам. Наиболее просты в применении модуль *turtle* — графическая библиотека, которая позволяет создавать простые рисунки и анимации с помощью черепашки, и библиотека визуализации данных *matplotlib*. В данной статье мы ограничимся примерами статической визуализации и не будем приводить примеры имитационного моделирования.

3. Примеры использования графических возможностей языка Python для создания учебных материалов

Пример 1. Декартова система координат. Рисунок елки.

Написать программу, которая рисует координатную плоскость и изображение елки в виде замкнутой ломаной линии и расставляет координаты вершин.

Программа для данного примера представлена в листинге 1, *результат работы программы* — на рисунке 1.

Результат работы программы можно использовать как на уроках математики при изучении темы «Декартова система координат», так и на уроках информатики при освоении понятий «экранная система координат», «декартова система координат», а также возможностей исполнителя Черепашка.

Пример 2. Значение логического выражения. Числовая прямая и множество решений задачи.

Написать программу, которая демонстрирует штриховкой на прямой решение задания 3 ОГЭ по информатике [6]:

Напишите наибольшее целое число X , для которого истинно высказывание:

$$\text{НЕ } (X \leq 14) \text{ И } (X \leq 18).$$

Решение представленной в условии задачи будет следующим: $X \in (14, 15, 16, 17, 18]$, т. е. $X = \{15, 16, 17, 18\}$. Наибольшее целое число — 18.

Программа для данного примера представлена в листинге 2, *результат работы программы* — на рисунке 2.

Код данной программы можно использовать (при небольшой доработке) для формирования визуального материала задач по математике по теме «Числовая плоскость. Множества, интервалы, отрезки».

Пример 3. Граф для линейного алгоритма.

Написать программу, которая рисует в виде цепи граф, вершины которого являются промежуточными значениями решения задания 5 ОГЭ по информатике [6]:

У исполнителя Гамма две команды, которым присвоены номера:

1. прибавь 3
2. умножь на b (b — неизвестное натуральное число; $b \geq 2$)

Выполняя первую из них, Гамма увеличивает число на экране на 3, а выполняя вторую, умножает это число на b . Программа для исполнителя Гамма — это последовательность номеров команд. Известно, что программа 11121 переводит число 3 в число 75. Определите значение b .

Программа для данного примера представлена в листинге 3, *результат работы программы* — на рисунке 3.

Код данной программы при доработке можно использовать для формирования визуального образовательного контента по теме «Графы». При небольших изменениях в коде программы можно получить множество иллюстративного материала для графического представления решения задач такого типа.

Пример 4. Круги Эйлера—Венна.

Написать программу, которая рисует круги Эйлера—Венна.

В листинге 4 представлен образец программы генерации визуального контента к задачам по теме «Круги Эйлера—Венна» [3], на рисунке 4 показан *результат работы программы*.

Результаты работы программы можно использовать для создания визуального образовательного контента для решения задания 8 ОГЭ по информатике [6] и при изучении дисциплин математического цикла.

Также можно для визуализации решения множества задач использовать возможности библиотеки визуализации данных *matplotlib*, библиотеки для работы с массивами и матрицами и выполнения высокоуровневых математических функций *numpy*, библиотеки для создания статистических графиков *seaborn*.

```

import turtle
t=turtle.Turtle()

t.pensize(2)
t.up()
t.goto(-200,200)
t.down()
t.goto(200,200)
t=turtle.Turtle()
t.hideturtle()
t.up()
t.goto(200,210)
t.down()
t.write('x',font=('arial',24))

for i in range(10):
    t.up()
    t.goto(-200+i*40,210)
    t.down()
    t.goto(-200+i*40,195)

for i in range(10):
    t.up()
    t.goto(-200+i*40,165)
    t.down()

t.write(i+11,font=('arial',17))

for i in range(28):
    t.up()
    t.goto(-200+i*10,220)
    t.down()
    t.goto(-190+i*10,200)
    for i in range(28):
        t.up()
        t.goto(-70+i*10,220)
        t.down()
        t.goto(-80+i*10,200)

    t.up()
    t.goto(80,200)
    t.down()
    t.pensize(3)
    t.dot(20)

    t.up()
    t.goto(-80,190)
    t.down()
    t.pensize(3)
    t.circle(10)

    t.up()
    t.goto(80,200)
    t.down()
    t.pensize(3)
    t.dot(10)
    t.screen.exitonclick()
    t.screen.mainloop()

```

Листинг 2. Решение примера 2

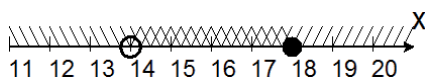


Рис. 2. Результат работы программы примера 2

Для их установки, если Python уже установлен, нужно использовать пакетный менеджер pip: нажать на клавиатуре Win+R (открыть утилиту *Выполнить*) и в появившейся командной строке записать:

```

pip install matplotlib
pip install numpy
pip install seaborn

```

После установки нужных модулей необходимо импортировать их в проект:

```

import matplotlib.pyplot as plt
import numpy as np
import seaborn as sns

```

Пример 5. Сердце.

Написать программу, которая строит график в форме сердца.

Формула для уравнения кривой второго порядка с параметром, которая имеет форму сердца:

$$\begin{cases} x = 16 \cdot \sin^3(t); \\ y = 13 \cdot \cos(t) - 5 \cdot \cos(2 \cdot t) - 2 \cdot \cos(3 \cdot t) - \cos(4 \cdot t). \end{cases}$$

Программа для данного примера представлена в листинге 5, результат работы программы — на рисунке 5.

Визуализацию можно использовать на уроках математики для представления учащимся возможностей использования тригонометрических функций.

4. Обсуждение возможностей применения языков программирования для формирования визуального обучающего контента

Рассмотренные выше примеры наглядно демонстрируют, что для создания визуальных моделей и формирования цифрового образовательного контента и иных образовательных материалов можно и нужно использовать широкие возможности графических пакетов и библиотек языков программирования.

Проблема наглядного представления учебной информации с точки зрения методики преподавания порождает новые формы методической деятельности в условиях изменений, происходящих в сфере образования в связи с активным использованием цифровых информационных технологий [5, 11]. Примеры, аналогичные представленным выше, можно давать учащимся на уроках в качестве заданий для самостоятельной работы, т. е. заданием будет не только решить задачу, но и написать программу, которая формирует графическую модель [2, 8, 10].

```

import turtle
t=turtle.Turtle()
t.hideturtle()
t.pensize(3)
s='11121'
s=list(s)
k1=3
kn=75
b1=3
t.speed(200)

for i in range (6):
    t.up()
    t.goto(-200+100*i,0)
    t.down()
    t.circle(25)

for i in range (5):
    t.up()
    t.goto(-175+100*i,25)
    t.down()
    t.fd(50)
    t.left(135)
    t.fd(10)
    t.backward(10)
    t.left(90)

t.fd(10)
t.backward(10)
t.left(135)
t.up()
t.goto(-207,11)
t.down()
t.write(k1,font=('arial',20))
t.up()
t.goto(288,11)
t.down()
t.write(kn,font=('arial',20))
for i in range (5):
    t.up()
    t.goto(-160+100*i,25)
    t.down()
    t.write('№'+s[i],font=('arial',13))
for i in range (5):
    t.up()
    t.goto(-163+100*i,0)
    t.down()
    if s[i]=='1': t.write('+'+str(b1),
        font=('arial',15))
    if s[i]=='2': t.write('*b',
        font=('arial',15))
t.screen.exitonclick()
t.screen.mainloop()

```

Листинг 3. Решение примера 3

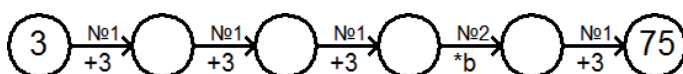


Рис. 3. Результат работы программы примера 3

```

import turtle
t=turtle.Turtle()
t.hideturtle()
t.pensize(3)
t.speed(200)
t.up()
t.goto(-50,0)
t.down()
t.circle(100)
t.up()
t.goto(50,0)
t.down()
t.circle(100)
t.up()
t.goto(-90,75)
t.down()
t.write('X',font=('arial',20))
t.up()

t.goto(0,75)
t.down()
t.write('Y',font=('arial',20))
t.up()
t.goto(75,75)
t.down()
t.write('Z',font=('arial',20))
t.up()
t.goto(-230,200)
t.down()
t.write('Спортсмены',font=('arial',20))
t.up()
t.goto(100,200)
t.down()
t.write('Певцы',font=('arial',20))
t.screen.exitonclick()
t.screen.mainloop()

```

Листинг 4. Решение примера 4

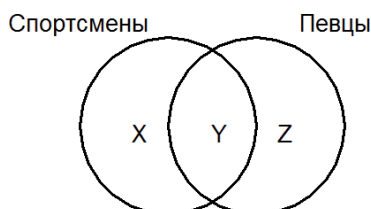


Рис. 4. Результат работы программы примера 4

```
import numpy as np
import matplotlib.pyplot as plt
fig=plt.figure()
t=np.arange(0,6.28,0.01)
x=(np.sin(t))**3

y=13*(np.cos(t))-5*(np.cos(2*t))
-2*(np.cos(3*t))-(np.cos(4*t))
plt.plot(x,y,color='green')
plt.show()
```

Листинг 5. Решение примера 5

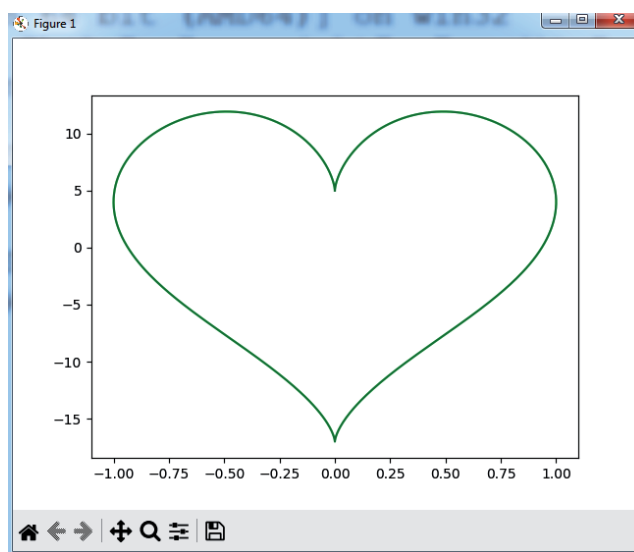


Рис. 5. Результат работы программы примера 5

5. Выводы

В статье показаны возможности языка программирования для создания цифрового образовательного контента и использования его в школьных курсах информатики, математики и других дисциплин. Представлен комплекс примеров для создания графических моделей по различным темам и разным школьным предметам. Предложенные подходы позволяют автоматизировать создание цифрового образовательного контента и использовать его как в создании наглядных учебных пособий, так и в качестве материала для обучения учащихся использованию графических возможностей языка Python. Визуальные образы формируют более полное и активное усвоение материала и успешное формирование личностных, метапредметных и предметных результатов учащихся.

Общедоступность программного обеспечения в виде языка программирования Python и возможностей его графических модулей и библиотек способствует применению рассмотренных методов широким кругом пользователей без ограничения со стороны программного обеспечения.

Список источников

1. Аранова С. В. К вопросу о принципах визуального представления учебной информации // Известия Российского государственного педагогического университета им. А. И. Герцена. 2017. №. 185. С. 96–102. EDN: XNGMKT.
2. Бешенков С. А., Акимов И. В. Визуализация как метод обучения программированию // Информатика и образование. 2017. № 10. С. 11–15. EDN: ZWQDKN.
3. Босова Л. Л. Информатика. 9 класс. Базовый уровень: учебник ФГОС. М.: Просвещение, 2023. 272 с.

4. Босова Л. Л., Филиппов В. И., Босова А. Ю. Программирование графики и анимации в курсе информатики основной школы // Информатика в школе. 2022. № 5. С. 5–15. EDN: LYUPWI. DOI: 10.32517/2221-1993-2022-21-5-5-15.

5. Бровка Н. В. Об использовании методики обучения школьников основам алгоритмизации программирования с применением визуализированных языков программирования // Международный конгресс по информатике: информационные системы и технологии (в образовании) — 2020. Сборник материалов Международного научного конгресса (Минск, 22–23 октября 2020 года). Минск: Белорусский государственный университет, 2020. С. 59–64. EDN: BGRQYD.

6. Демоверсии, спецификации, кодификаторы // ФИПИ. <https://fipi.ru/oge/demoversii-specifikacii-kodifikatory>

7. Краснопахтова Л. И., Танкаян А. И. Визуализация как средство эффективного повышения уровня восприятия информации и обеспечения интенсификации обучения // Интерактивная наука. 2018. №. 6 (28). С. 34–36. EDN: USBKLF. DOI: 10.21661/r-472178.

8. Моглан Д. В. Дидактический потенциал использования систем визуализации алгоритмов в процессе обучения программированию // Открытое образование. 2019. № 23 (2). С. 31–41. EDN: JEDGGF. DOI: 10.21686/1818-4243-2019-2-31-41.

9. Полякова Е. В. Применение способов и методов визуального мышления в современном образовании // Известия ЮФУ. Технические науки. 2012. № 10 (135). С. 120–124. EDN: PEVLOJ.

10. Раджабова Ф. М., Гусейнова М. М., Исакова М. Н. Визуализация как метод обучения программированию // Вопросы устойчивого развития общества. 2022. № 5. С. 525–531. EDN: QDCQBY.

11. Савельева С. В., Валеева И. Х. О возможности решения проблемы визуализации учебной информации в процессе изучения дисциплины «Информатика» // Инновационное развитие профессионального образования. 2019. Т. 22. №. 2. С. 50–55. EDN: TKUEFC.