



DOI: 10.32517/2221-1993-2022-21-6-68-80

В. К. Маркелов, О. А. Завьялова

Шуйский филиал Ивановского государственного университета, г. Шуя, Ивановская область, Россия

ИНСТРУМЕНТАРИЙ ОБЛАЧНОЙ СРЕДЫ GOOGLE COLABORATORY ДЛЯ ОБУЧЕНИЯ ПРОГРАММИРОВАНИЮ И ПОДГОТОВКИ К ЕГЭ ПО ИНФОРМАТИКЕ

Аннотация

Одной из облачных технологий, которую можно использовать для обучения программированию на базе языка программирования Python, является облачный сервис Google Colaboratory. Данный сервис может использоваться учителями как при изучении отдельных тем раздела «Алгоритмы и программирование» в рамках школьного курса информатики, так и при подготовке обучающихся к заданиям по программированию ЕГЭ по информатике.

В статье рассматриваются возможности облачного сервиса Google Colaboratory для создания интерактивных блокнотов на примере интерактивного блокнота и практической работы по теме «Цикл с предусловием», а также интерактивных блокнотов «Задание № 24 ЕГЭ по информатике» для обучающихся и для учителя. Обосновывается применимость данных блокнотов для обучения программированию на базе языка Python в рамках школьного курса информатики и для подготовки обучающихся к заданиям по программированию ЕГЭ по информатике.

Ключевые слова: программирование, интерактивный блокнот, методика обучения программированию, Google Colaboratory, Python.

1. Введение

Методика обучения программированию в рамках школьного курса информатики нуждается в постоянном обновлении и совершенствовании используемых методов, технологий и средств. В качестве одного из таких средств могут применяться различные облачные технологии. При этом в рамках образовательного процесса облачные технологии обычно используются для

предоставления доступа учащимся к учебно-методическим материалам, для файлообмена между обучающимся и учителем, а также между самими обучающимися с целью обмена опытом и совместной работы над учебными проектами.

Под облачными технологиями понимаются системы обработки данных, в которых компьютерные ресурсы и мощности предоставляются пользователю как интернет-сервис [9].

Контактная информация

Маркелов Валерий Константинович, студент 2-го курса магистратуры по направлению подготовки 44.04.01 «Педагогическое образование» (профиль «Информационные технологии в профессиональной деятельности педагога»), Шуйский филиал Ивановского государственного университета, г. Шуя, Ивановская область, Россия; *адрес:* 155908, Россия, Ивановская область, г. Шуя, ул. Кооперативная, д. 24; *e-mail:* v. a.l. e.m. a.r.k@yandex.ru

Завьялова Ольга Алексеевна, канд. пед. наук, доцент кафедры математики, информатики и методики обучения, Шуйский филиал Ивановского государственного университета, г. Шуя, Ивановская область, Россия; *адрес:* 155908, Россия, Ивановская область, г. Шуя, ул. Кооперативная, д. 24; *e-mail:* ooolga30@yandex.ru

V. K. Markelov, O. A. Zavyalova

Shuya branch of Ivanovo State University, Shuya, Ivanovo Region, Russia

TOOLS OF THE GOOGLE COLABORATORY CLOUD ENVIRONMENT FOR PROGRAMMING TRAINING AND THE PREPARATION FOR THE UNIFIED STATE EXAM IN INFORMATICS

Abstract

One of the cloud technologies that can be used to teach programming based on the Python programming language is the Google Colaboratory cloud service. This service can be used by teachers both when teaching certain themes of the section "Algorithms and programming" as part of a school informatics course, and when preparing students for the tasks in programming of the Unified State Exam in informatics.

The article discusses the possibilities of the Google Colaboratory cloud service for creating interactive notebooks using the example of an interactive notebook and practical work on the theme "A precondition loop", as well as interactive notebooks "Task No. 24 of the Unified State Exam in informatics" for students and teachers. The applicability of these notebooks for programming training based on the Python language as part of a school informatics course and for preparing students for tasks in programming of the Unified State Exam in informatics is substantiated.

Keywords: programming, interactive notebook, methodology of programming training, Google Colaboratory, Python.

Современный урок информатики допускает параллельную работу обучающихся с различными языками программирования, ставшую возможной благодаря наличию облачных сервисов, позволяющих изучать программирование без установки программного обеспечения на свой компьютер [4]. Одним из основных языков программирования в рамках школьного курса информатики является Python. Python — это язык программирования высокого уровня, простой синтаксис которого позволяет учащимся быстро освоить базовые алгоритмические структуры (следование, ветвление, выбор, повторение) и акцентировать основное внимание на оттачивании навыков программирования, при этом улучшенные отчеты об ошибках способствуют развитию у обучающихся умения находить ошибки в программе и самостоятельно их исправлять [7].

Использование облачных сервисов в образовательном процессе создает условия для естественного развития познавательных и творческих способностей обучающихся, развивает коммуникативные и личностные универсальные учебные навыки, а также способности к планированию и управлению своей и совместной деятельностью [5]. Кроме того, применение данных сервисов позволяет снизить затраты на процесс обучения, повысить качество подготовки учащихся за счет оперативного изменения учебных планов, формирования и предоставления совместного доступа к обучающим материалам [6].

Одним из облачных сервисов, который можно использовать для обучения программированию в школе на базе языка Python, является облачный сервис Google Colaboratory, который был разработан корпорацией Google.

Google Colaboratory, или Google Colab (<https://colab.research.google.com/>) — это бесплатная интерактивная облачная среда, которая используется в профессиональной разработке для анализа данных и машинного обучения на базе языка Python 3.8.16. Облачная среда Google Colaboratory стимулирует улучшение навыков программирования на Python, а также позволяет реализовывать алгоритмы углубленного обучения с использованием популярных библиотек, таких как Keras, TensorFlow, PyTorch и OpenCV [10].

Данный сервис позволяет специалистам в области цифровых технологий [8]:

- совместно анализировать структурированные и неструктурированные данные огромных объемов (Big Data);
- проектировать и тестировать новые модели машинного обучения;
- разрабатывать нейронные сети;
- осуществлять эксперименты в области искусственного интеллекта;
- моделировать и имитировать различные процессы;
- разрабатывать программное обеспечение.

Основу облачной среды Google Colaboratory составляет интерактивный блокнот Jupyter Notebook, в котором присутствует возможность работы как со статическим содержимым (текст, изображения, видео), так и с программным кодом на Python.

Сегодня у большинства обучающихся есть различные электронные устройства, поэтому облачные технологии могут активно использоваться в образовании, а их внедрение имеет определенные преимущества. Так,

применение облачных технологий открывает учителю широкие возможности для организации уровневой дифференциации и индивидуального подхода, а также появляется возможность создания учебного модуля с упражнениями различного уровня сложности [3].

Облачная среда Google Colaboratory может использоваться как учителями информатики, так и обучающимися в рамках изучения языка программирования Python. Привлекательность применения в образовательном процессе данного облачного сервиса состоит в том, что, имея только учетную запись Google и подключение к интернету, учителя могут использовать облачную среду Google Colaboratory для создания различных интерактивных средств обучения к урокам раздела «Программирование» в рамках школьного курса информатики, в частности, для создания обучающих интерактивных блокнотов. *Под интерактивными средствами обучения понимаются программные, аппаратно-программные и технические средства и устройства, которые функционируют на базе микропроцессорной и вычислительной техники и обеспечивают обучение в диалоговом взаимодействии пользователя с компьютером* [2].

Следует отметить, что программирование составляет основу содержательной линии «Алгоритмы и программирование», изучаемой на протяжении всего школьного курса информатики, поэтому задачи по программированию являются одним из тематических блоков Единого государственного экзамена по информатике. Актуальным способом подготовки к решению задач по программированию ЕГЭ является как непосредственное изучение языка программирования, так и изучение основных понятий языка с решением достаточного количества соответствующих задач, покрывающих основные идеи так называемых школьных алгоритмов [1]. Благодаря возможностям облачной среды Google Colaboratory интерактивные блокноты могут также использоваться и при подготовке обучающихся к решению задач по программированию ЕГЭ.

Целью данной статьи является описание инструментария и возможностей бесплатной интерактивной облачной среды Google Colaboratory, а также демонстрация примеров готовых интерактивных блокнотов — как для обучения программированию на базе языка Python в рамках школьного курса информатики, так и для подготовки обучающихся к решению задач по программированию единого государственного экзамена.

2. Инструментарий облачной среды Google Colaboratory для создания и использования обучающих интерактивных блокнотов при обучении программированию в школе

Проанализируем инструментарий облачной среды Google Colaboratory на примере обучающего интерактивного блокнота «Цикл с предусловием»:

https://colab.research.google.com/drive/13QBT_WRZimZFTpitJK3mUQumZ12D0whq

Этот блокнот может применяться в рамках изучения соответствующей темы раздела «Алгоритмы и программирование».

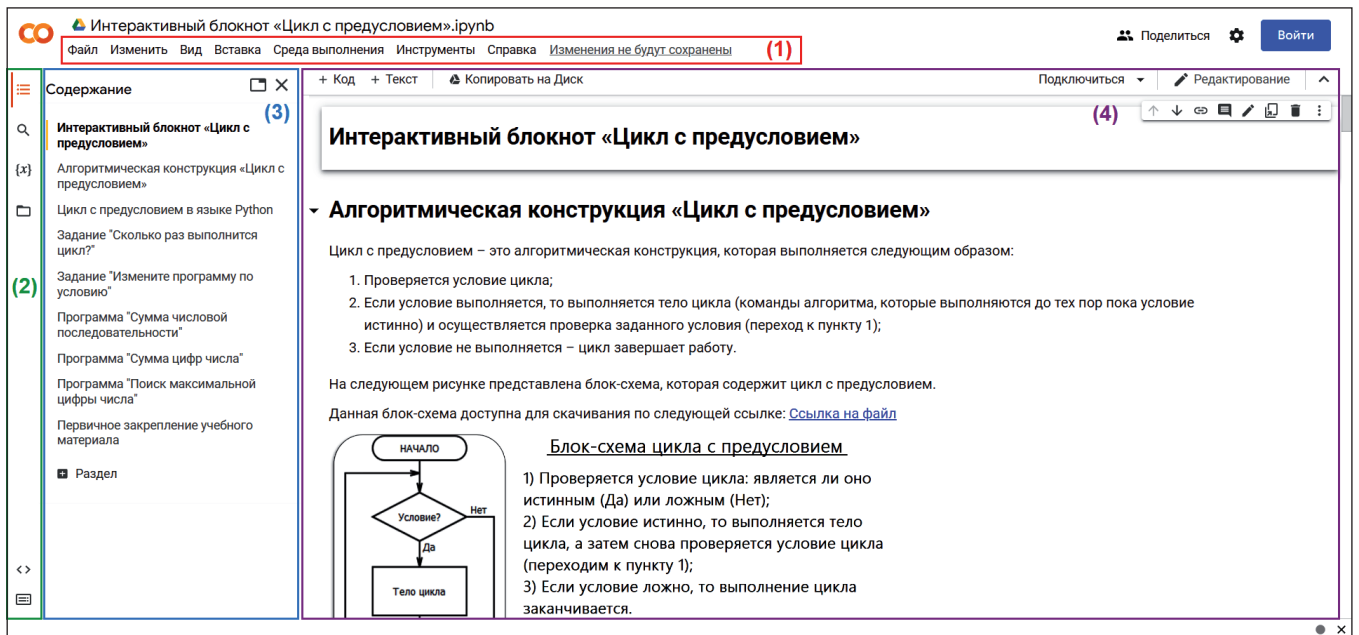


Рис. 1. Основные элементы облачной среды Google Colaboratory на примере интерактивного блокнота «Цикл с предусловием»

Рассмотрим основные элементы облачной среды Google Colaboratory (рис. 1). В верхней части окна расположено меню (1), над которым стандартно располагается название интерактивного блокнота. В левой части окна находится боковая панель (2). При нажатии на самую верхнюю кнопку данной боковой панели открывается колонка *Содержание* (3), которая позволяет перемещаться между разделами и подразделами интерактивного блокнота. Данную колонку можно закрыть при помощи соответствующей кнопки на боковой панели, подсвеченной оранжевым цветом. Оставшаяся часть окна представляет собой рабочую область среды (4) с различным интерактивным содержимым, которое расположено в текстовых ячейках (текст, изображения, видео) и кодовых ячейках (программы или фрагменты кода на языке Python).

Для внесения изменений в интерактивный блокнот необходимо войти в свою учетную запись Google и скопировать блокнот на свой Google Диск при помощи кнопки *Копировать на Диск*. Если не войти в учетную запись, внесенные в блокнот изменения будут удалены после закрытия страницы с интерактивным блокнотом. Об этом также информирует запись «Изменения не будут сохранены», расположенная справа от меню облачной среды.

Интерактивный блокнот «Цикл с предусловием» состоит из восьми разделов:

- «Алгоритмическая конструкция «Цикл с предусловием»»;
- «Цикл с предусловием в языке Python»;
- «Задание «Сколько раз выполнится цикл?»»;

- «Задание «Измените программу по условию»»;
- «Программа «Сумма числовой последовательности»»;
- «Программа «Сумма цифр числа»»;
- «Программа «Поиск максимальной цифры числа»»;
- «Первичное закрепление учебного материала».

Заголовок раздела «Алгоритмическая конструкция «Цикл с предусловием»» в **режиме редактирования** представлен на рисунке 2. Чтобы перейти в данный режим, достаточно дважды нажать на заданную ячейку. Для создания такого заголовка используется текстовая ячейка для заголовка раздела, которую можно создать при помощи команды меню *Вставка, Ячейка для заголовка раздела*. При создании ячейки открывается окно редактирования ячейки, в котором можно превратить раздел в подраздел при помощи кнопки (1), сделать текст заголовка жирным (2) или курсивным (3). Остальные кнопки используются в основном для работы с текстовыми ячейками. Кроме того, каждая ячейка имеет управляющее меню (4), с помощью команд которого можно:

- перемещать ячейки по блокноту (стрелки вверх-вниз);
- получить ссылку на ячейку (кнопка со скрепкой);
- добавить комментарий (кнопка со значком сообщения);
- завершить изменения в ячейке (кнопка с зачеркнутым карандашом);
- удалить ячейку (кнопка с изображением корзины).

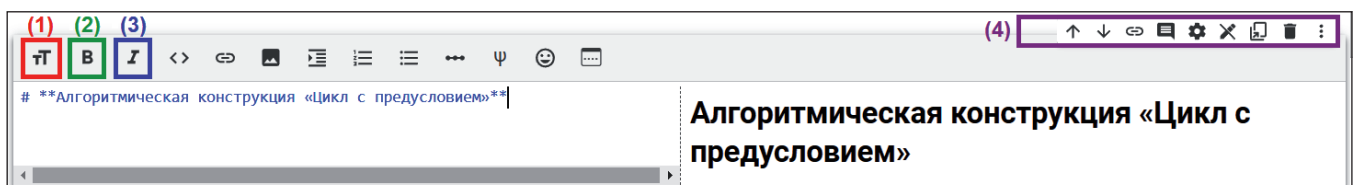


Рис. 2. Редактор ячейки для заголовка раздела «Алгоритмическая конструкция «Цикл с предусловием»»

Цикл с предусловием в языке программирования Python выглядит следующим образом:

```

while <условие>:
    <команда 1>
    <команда 2>

```

где `<условие>` - это условие цикла, а `<команда1>`, `<команда2>` - это команды тела цикла.

В качестве примера рассмотрим программу, которая выводит слово "Привет!" на экран 5 раз.

1. В программе используется переменная `**i**` - так называемый "счетчик", которой присваивается значение `*i = 0*`.
2. В цикле задаётся условие `*i < 5*`, и если оно выполняется, то на экран выводится слово "Привет!", значение переменной-счетчика `**i**` увеличивается на `*1*`, а затем снова проверяется условие `*i < 5*` (то есть осуществляется переход к следующей итерации, итерация в данном случае - это повторение команд тела цикла).
3. С каждой итерацией значение переменной `*i*` будет увеличиваться на `*1*`, следовательно, тело цикла выполнится `*5*` раз и на экран `*5*` раз будет выведено слово "Привет!". Поэтому в программе задается условие цикла `*i < 5*`. Когда значение переменной изменится на `*i = 5*`, цикл завершит работу.

> Ниже представлена программа на языке программирования Python. Запустите её, используя интерактивный блокнот.

Цикл с предусловием в языке программирования Python выглядит следующим образом:

```

while <условие>:
    <команда 1>
    <команда 2>

```

где `<условие>` - это условие цикла, а `<команда1>`, `<команда2>` - это команды тела цикла.

В качестве примера рассмотрим программу, которая выводит слово "Привет!" на экран 5 раз.

1. В программе используется переменная `i` - так называемый "счетчик", которой присваивается значение `i = 0`.
2. В цикле задаётся условие `i < 5`, и если оно выполняется, то на экран выводится слово "Привет!", значение переменной-счетчика `i` увеличивается на `1`, а затем снова проверяется условие `i < 5` (то есть осуществляется переход к следующей итерации, итерация в данном случае - это повторение команд тела цикла).
3. С каждой итерацией значение переменной `i` будет увеличиваться на `1`, следовательно, тело цикла выполнится `5` раз и на экран `5` раз будет выведено слово "Привет!". Поэтому в программе задается условие цикла `i < 5`. Когда значение переменной изменится на `i = 5`, цикл завершит работу.

Ниже представлена программа на языке программирования Python. Запустите её, используя интерактивный блокнот.

Рис. 3. Редактор текстовой ячейки раздела «Цикл с предусловием в языке Python»

В разделе «Алгоритмическая конструкция “Цикл с предусловием”» рассматривается алгоритм выполнения цикла с предусловием и приводится способ записи данной алгоритмической конструкции с использованием блок-схемы. В свою очередь, в разделе «Цикл с предусловием в языке Python» рассматривается структура цикла с предусловием в языке программирования Python и приводится простейший пример программы с использованием данного цикла, которая пять раз выводит на экран слово «Привет!»

Для отображения информации в облачной среде Google Colaboratory существует два основных типа ячеек: текстовые и кодовые.

В качестве примера рассмотрим **текстовую ячейку** раздела «Цикл с предусловием в языке Python» в режиме редактирования (рис. 3). Для создания такой ячейки можно использовать кнопку *Текст* в верхней части рабочей области среды или команду меню *Вставка, Текстовая ячейка*. При создании ячейки открывается окно редактирования ячейки, в котором, так же как и для ячейки заголовка раздела, можно превратить раздел в подраздел при помощи кнопки (1), сделать текст заголовка жирным (2) или курсивным (3), оформить текст в виде программного кода, используя кнопку (4). Кнопка (5) позволяет вставить в ячейку ссылку, а кнопка (6) — изображение. Кроме того, текст можно выделить при помощи отступа (7), который дополнительно обозначается серой вертикальной линией. В ячейке также можно использовать маркированные (8) и нумерованные (9) списки, горизонтальную линию (10), математические формулы

в формате LaTeX (11) и эмодзи (12). По умолчанию окно редактирования ячейки при помощи специальной разделительной линии отделяется от окна предварительного просмотра (13), в котором можно посмотреть, как будет выглядеть ячейка после внесения изменений. Положение данного окна можно изменить (справа или снизу от окна редактирования) при помощи кнопки (14).

Следует отметить, что использование кнопок для редактирования текста не является обязательным. Для превращения текста в заголовок раздела может использоваться символ «#». Чтобы сделать текст жирным, достаточно заключить его между символами «**», а чтобы сделать текст курсивным — между символами «*». Символ «*», отделенный пробелом, может использоваться для создания маркированного списка, а цифра с точкой, отделенная пробелом, — для создания нумерованного списка. В свою очередь, для создания отступа применяется символ «>», а для создания горизонтальной линии — символы «---». При этом отступы, так же как и горизонтальные линии, должны отделяться от остального текста отдельной строкой.

Рассмотрим возможности добавления ссылок и изображений в облачной среде Google Colaboratory. Для этого в качестве примера рассмотрим текстовую ячейку раздела «Алгоритмическая конструкция “Цикл с предусловием”» (рис. 4).

При нажатии на кнопку добавления ссылки предлагается создать ссылку в формате [Текст ссылки](https://link.com), где `https://link.com` — некоторая ссылка.

Чтобы добавить изображение в текстовую ячейку, необходимо нажать на кнопку добавления изображения, выбрать изображение со своего компьютера и нажать на кнопку *Открыть*. При этом в текстовую ячейку добавляется конструкция формата [Имя файла](Ссылка на изображение). Следует отметить, что при загрузке изображения с компьютера ссылка может быть очень длинной, поскольку изображение загружается на серверы Google. Кроме того, такие изображения загружаются с оригинальными размерами, без возможности их изменения. Однако редактор текстовых ячеек поддерживает html-теги, поэтому для добавления изображений рекомендуется использовать html-тег «img».

Например, тег: ``

позволит создать изображение размером 500 на 300 пикселей (вместо «ссылка на изображение» необходимо указать ссылку, которая появится после загрузки изображения с компьютера, либо использовать ссылку из сети Интернет). Для создания ссылок также можно использовать команду:

`Ссылка`,

где вместо «http://link.com» находится ссылка.

Кроме текстовых ячеек, интерактивные блокноты также могут содержать **кодовые ячейки**, содержащие

фрагменты кода или программы на языке программирования Python. Интерактивные блокноты Google Colaboratory позволяют использовать в кодовых ячейках все возможности языка Python версии 3.8.16, в том числе стандартные библиотеки языка Python (math, random и т. д.), а недостающие библиотеки можно импортировать при помощи команды «!pip install название библиотеки» (библиотеки с поддержкой графического интерфейса, такие как tkinter, не поддерживаются интерактивными блокнотами Google Colaboratory).

В качестве примера рассмотрим кодовую ячейку раздела «Цикл с предусловием в языке Python», которая представлена на рисунке 5. В ней содержится программа «Привет», которая пять раз выводит на экран слово «Привет!». Для того чтобы запустить данную программу, необходимо нажать на кнопку запуска (1), которая появляется при наведении курсора на символы «[]». При этом под кодовой ячейкой появятся результаты работы программы, левее ячейки — зеленая галочка (если программа выполнена успешно, без ошибок), а под ней — время выполнения программы в секундах (2).

Интерактивный блокнот «Цикл с предусловием» содержит различные типы практических заданий для закрепления полученных знаний по соответствующей теме.

Задание «Измените программу по условию» предлагает изменить заданные программы на языке Python так, чтобы цикл выполнялся определенное количество

Цикл с предусловием – это алгоритмическая конструкция, которая выполняется следующим образом:

1. Проверяется условие цикла;
2. Если условие выполняется, то выполняется тело цикла (команды алгоритма, которые выполняются до тех пор, пока условие истинно) и осуществляется проверка заданного условия (переход к пункту 1);
3. Если условие не выполняется – цикл завершает работу.

На следующем рисунке представлена блок-схема, которая содержит цикл с предусловием.

Данная блок-схема доступна для скачивания по следующей ссылке: [Ссылка на файл] (https://drive.google.com/file/d/1NzprJwjsW63HjzDkLk_ZlP1039W5jvstD/view?usp=sharing)

```
 b:
        a = a + b
        print(a)

```

Программа №4.

```

[ ] a = 4
    b = 7
    while a < b:
        b = a - b
        print(b)

```

Программа №5.

```

[ ] a = 4
    b = 7
    while a < b:
        a = a - 1
        print(a)

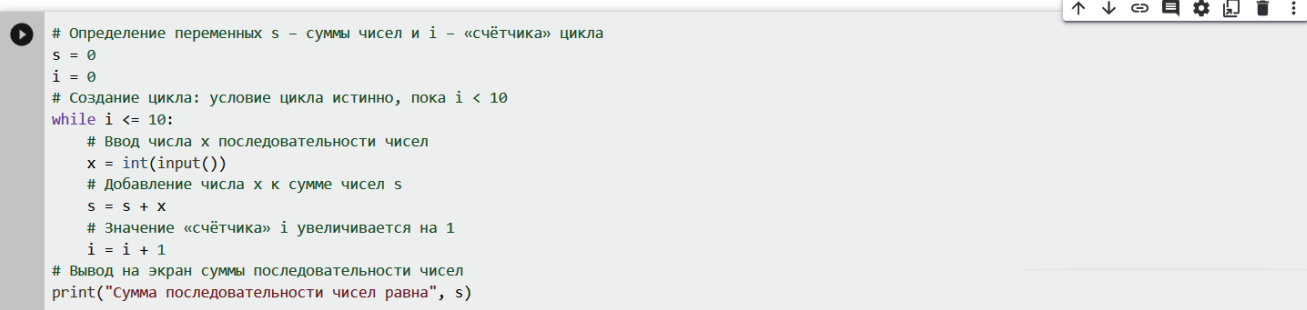
```

Рис. 6. Задание «Сколько раз выполнится цикл?»

▼ Программа "Сумма числовой последовательности"

Рассмотрим возможности циклов с предусловием для написания различных программ для работы с числами. Одними из самых простых задач с использованием данной алгоритмической конструкции являются алгоритмы обработки последовательностей чисел. В качестве примера такой программы рассмотрим следующую задачу.

Напишите программу на языке программирования Python, которая получает на вход последовательность из **10** целых чисел, вводимых с клавиатуры, находит их сумму и выводит её на экран.



```
# Определение переменных s – суммы чисел и i – «счётчика» цикла
s = 0
i = 0
# Создание цикла: условие цикла истинно, пока i < 10
while i <= 10:
    # Ввод числа x последовательности чисел
    x = int(input())
    # Добавление числа x к сумме чисел s
    s = s + x
    # Значение «счётчика» i увеличивается на 1
    i = i + 1
# Вывод на экран суммы последовательности чисел
print("Сумма последовательности чисел равна", s)
```

Для решения данной задачи необходимо задать переменные **s = 0** – сумма чисел, вводимых с клавиатуры и **i = 0** – "счетчик" цикла, который будет подсчитывать, сколько раз выполнялся цикл и, соответственно, сколько чисел было введено с клавиатуры.

Далее задается цикл с предусловием, который выполняется до тех пор, пока истинно условие $i \leq 10$, т.е. пока пользователь не ввел с клавиатуры 10 чисел.

Если условие выполняется, программа получает введенное с клавиатуры число **x** и добавляет его к сумме чисел **s**. Затем значение "счётчика" цикла **i** увеличивается на 1, чтобы подсчитать введенное с клавиатуры число.

После того как была найдена сумма 10 чисел, она выводится на экран.

Подумайте, как можно изменить данную программу, чтобы она находила произведение 5 чисел, вводимых с клавиатуры. Какое начальное значение должна иметь переменная произведения чисел? Какое условие необходимо задать в цикле программы?

Рис. 7. Программа «Сумма числовой последовательности»

вательность из 10 целых чисел, вводимых с клавиатуры, находит их сумму и выводит ее на экран;

- «Сумма цифр числа» — программа получает на вход натуральное число n и выводит на экран произведение его цифр;
- «Поиск максимальной цифры числа» — программа получает на вход натуральное число x и выводит на экран его максимальную цифру.

Данные программы обучающийся может запустить непосредственно в блокноте, без необходимости устанавливать среду программирования IDLE Python на компьютер или использовать онлайн-среды с возможностью написания программ на языке программирования Python. Таким образом, интерактивный блокнот «Цикл с предусловием» можно использовать на уроке открытия новых знаний, обретения новых умений и навыков непосредственно на этапе изучения нового материала.

В разделе «Первичное закрепление учебного материала» для закрепления полученных знаний обучающимся предлагается выполнить **практическую работу «Цикл с предусловием»**, представленную в форме интерактивного блокнота:

<https://colab.research.google.com/drive/12nrTVMBPFyB1oO-3PAEPaKbp9xcpUiiJ?usp=sharing>.

Данная практическая работа направлена на проверку знаний обучающихся по теме «Цикл с предусловием»,

полученных в рамках изучения теоретического материала по соответствующей теме, который был рассмотрен ранее. В работе обучающемуся необходимо выполнить **пять практических заданий**:

- «Исправь программу» (задание на исправление программы таким образом, чтобы цикл в программе выполнялся заданное число раз);
- «Повтори фразу» — задание на написание программы, которая получает на вход строку s , а затем выводит ее на экран четыре раза;
- «Произведение числовой последовательности» — задание на написание программы, которая получает на вход натуральное число n и последовательность из n целых чисел, вводимых с клавиатуры, находит их произведение и выводит его на экран;
- «Произведение цифр числа» — задание на написание программы, которая получает на вход натуральное число x и выводит на экран число m — произведение цифр числа x ;
- «Минимальная цифра числа» — задание на написание программы, которая получает на вход натуральное число x и выводит на экран число m — минимальную цифру числа x .

Первое задание практической работы представлено на рисунке 8. Обучающийся имеет возможность выполнить практическое задание непосредственно в блокноте, написав программу в соответствующем поле. Кроме того, он может проверить правильность работы написан-



Практическая работа «Цикл с предусловием»

Инструкция: познакомьтесь с интерактивным блокнотом "Цикл с предусловием" и выполните следующие практические задания. Прежде чем приступать к заданиям, нажмите на кнопку "Копировать на диск", чтобы сохранить результаты работы.

Задание №1. Исправь программу

Постановка задания: Ниже представлены 4 программы на языке программирования Python.

Измените одну команду программы на языке Python так, чтобы:

- для программы А: цикл выполнился 3 раза;
- для программы Б: цикл выполнился 1 раз;
- для программы В: цикл не выполнялся ни разу;
- для программы Г: цикл выполнялся бесконечно;

<pre>a = 5 b = 11 while a < b: a = a + 1 print(a)</pre> <p style="text-align: right; background-color: #800000; color: white; padding: 2px; border-radius: 5px; display: inline-block;">А</p>	<pre>a = 20 b = 4 while a > b: a = a - b print(a)</pre> <p style="text-align: right; background-color: #800000; color: white; padding: 2px; border-radius: 5px; display: inline-block;">Б</p>	<pre>a = 8 b = 7 while a >= b: b = b + 1 print(b)</pre> <p style="text-align: right; background-color: #800000; color: white; padding: 2px; border-radius: 5px; display: inline-block;">В</p>	<pre>a = 18 b = 34 while a <= b: a = a + b print(a)</pre> <p style="text-align: right; background-color: #800000; color: white; padding: 2px; border-radius: 5px; display: inline-block;">Г</p>
--	--	--	--

Задание может иметь несколько различных решений!

```
# Исправьте программу А, которая представлена ниже
a = 5
b = 11
while a < b:
    a = a + 1
print(a)
```

```
[ ] # Исправьте программу Б, которая представлена ниже
a = 20
b = 4
while a > b:
    a = a - b
print(a)
```

```
[ ] # Исправьте программу В, которая представлена ниже
a = 8
b = 7
while a >= b:
    b = b + 1
print(b)
```

```
[ ] # Исправьте программу Г, которая представлена ниже
a = 18
b = 34
while a <= b:
    a = a + b
print(a)
```

Рис. 8. Задание «Исправь программу» практической работы «Цикл с предусловием»

ной им программы, нажав на кнопку запуска, которая появляется при наведении курсора на символы «[]», и запустив программу. Таким образом, практическую работу «Цикл с предусловием» можно использовать при изучении соответствующей темы раздела «Алгоритмы и программирование» как на уроке открытия новых знаний, обретения новых умений и навыков непосредственно на этапе первичного закрепления нового материала, так и на уроке комплексного применения знаний и умений на этапе актуализации знаний.

Для того чтобы иметь возможность **отправить результат своей работы учителю**, обучающемуся сначала необходимо зарегистрироваться или войти в свою

учетную запись Google, нажать на кнопку *Копировать на диск* и только затем приступить к практическим заданиям работы, в противном случае после перезагрузки страницы с интерактивным блокнотом результаты работы будут утеряны. После выполнения заданий практической работы обучающийся может воспользоваться кнопкой *Поделиться* для того, чтобы создать ссылку на интерактивный блокнот.

Окно предоставления доступа к интерактивному блокноту представлено на рисунке 9. Чтобы создать ссылку на блокнот, необходимо во вкладке *Общий доступ* выбрать пункт *Все, у кого есть ссылка* и нажать на появившуюся кнопку *Копировать ссылку*. Кроме того,

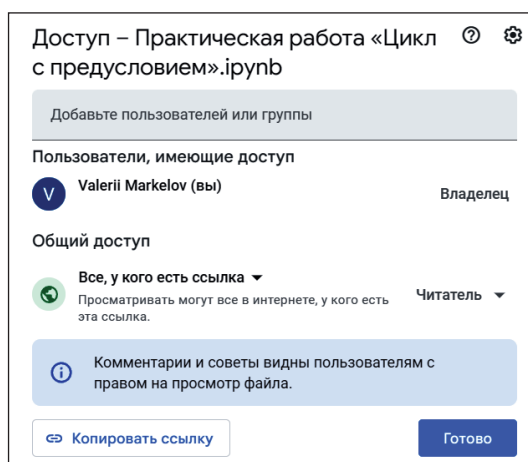


Рис. 9. Окно предоставления доступа к интерактивному блокноту



Рис. 10. Отображение комментария к ячейке интерактивного блокнота

используя текстовое поле *Добавьте пользователя или группы*, можно предоставить доступ к блокноту по адресу электронной почты преподавателя (если он привязан к учетной записи Google).

При проверке практической работы учитель может воспользоваться **функцией добавления комментариев к ячейке**. В качестве примера рассмотрим комментарий к заданию «Повтори фразу». Чтобы создать комментарий, необходимо в управляющем меню выбранной ячейки нажать на кнопку *Добавить комментарий* (кнопка со значком сообщения), в появившемся окне написать комментарий и нажать на кнопку *Комментировать*. Комментарий будет отображаться справа от ячейки, к которой он относится (рис. 10).

Учителя могут как использовать готовые обучающие блокноты интерактивной облачной среды Google Colaboratory, так и создавать собственные интерактивные блокноты для обучения программированию в рамках школьного курса информатики на базе языка Python.

3. Интерактивные блокноты как инструмент подготовки обучающихся к решению заданий по программированию ЕГЭ по информатике

Облачная среда Google Colaboratory может использоваться как в рамках изучения отдельных тем раздела «Алгоритмы и программирование», так и при подготовке обучающихся к ЕГЭ по информатике, в частности, в контексте заданий по программированию, направленных

на проверку умения создавать собственные программы. Одной из особенностей таких заданий является необходимость обработки информации из одного или нескольких текстовых файлов.

Рассмотрим возможности использования Google Colaboratory для обучения решению данных заданий на базе языка программирования Python на примере задания № 24 ЕГЭ по информатике.

Интерактивный блокнот «Задание № 24 ЕГЭ по информатике» (для обучающихся) (рис. 11) доступен по ссылке:

<https://colab.research.google.com/drive/1NyAca9zOlpXBztLiYyWS9-bAcl8u2GF5?usp=sharing>

Данный блокнот состоит из четырех разделов:

- «Теоретические сведения для решения задач»;
- «Примеры экзаменационных заданий»;
- «Список тренировочных заданий»;
- «Инструкция по отправке решения».

Раздел «Теоретические сведения для решения задач» содержит теоретические сведения, которые могут помочь обучающимся при решении задания № 24 ЕГЭ по информатике. В частности, в данном разделе рассматривается алгоритмическая конструкция для работы с файлами в языке программирования Python и приводятся различные методы работы со строками.

В разделе «Примеры экзаменационных заданий» представлены алгоритмы решения различных типов экзаменационных заданий № 24, в том числе из открытого банка заданий ЕГЭ.

Рассмотрим один из примеров экзаменационного задания (рис. 12). В данной задаче представлен тексто-

Интерактивный блокнот «Задание №24 ЕГЭ по информатике» (для обучающихся)

Файл Изменить Вид Вставка Среда выполнения Инструменты Справка Изменения не будут сохранены

Поделиться Войти

Содержание

Интерактивный блокнот «Задание №24 ЕГЭ по информатике»

- Теоретические сведения для решения задач
- Примеры экзаменационных заданий
- Список тренировочных заданий
- Инструкция по отправке решения
- Раздел

+ Код + Текст Копировать на Диск Подключиться Редактирование

Интерактивный блокнот «Задание №24 ЕГЭ по информатике»

Задание №24 в едином государственном экзамене (ЕГЭ) по информатике проверяет умение обучающегося создавать собственные программы (10-20 строк) для обработки символьной информации.

Максимальный балл за выполнение задания: 1 балл.
Примерное время выполнения задания: 18 минут.

Теоретические сведения для решения задач

Прежде всего для решения задачи необходимо получить символьную информацию (одну или несколько строк) из текстового файла.

В языке программирования Python получить символьную информацию из файла **file.txt** (которая записана в одну строку) в переменную **s** можно следующим образом:

```
with open("file.txt", "r") as f:
    s = f.readline()
```

Для этого в Python используется алгоритмическая конструкция **with-as**, которая при помощи команды **open** открывает файл **file.txt** в режиме чтения ("r" - аргумент команды open, указывающий открытие в режиме чтения).

Рис. 11. Интерактивный блокнот «Задание № 24 ЕГЭ по информатике» (для обучающихся)

Интерактивный блокнот «Задание №24 ЕГЭ по информатике» (для обучающихся)

Файл Изменить Вид Вставка Среда выполнения Инструменты Справка Изменения сохранены

Комментировать Поделиться Войти

Файлы

- sample_data
- 24-01.txt (4)

Скачать (2)

- Переименовать файл
- Удалить файл
- Скопировать путь
- Обновить (5)

+ Код + Текст

ОЗУ Диск Редактирование

Примеры экзаменационных заданий

Пример №1. Текстовый файл состоит из одной строки, которая содержит только буквы латинского алфавита. Определите, сколько раз в исходном файле встречается символ, стоящий в файле на первом месте. Для выполнения этого задания следует написать программу.

Текстовый файл задания доступен по следующей ссылке: <https://drive.google.com/file/d/1TOY-PuE03d5iyDTPQuM4VvGDit27-4NI/view?usp=sharing> (1)

Решение: Для проверки правильности решения задачи скачайте файл **24-01.txt**, для этого выберите пункт "Файлы" на боковой панели окна облачной среды и нажмите на кнопку "Загрузить в сессионное хранилище" (значок документа со стрелкой вверх). Обратите внимание, что ссылка на файл указывается в формате [/content/24-01.txt](#), где **24-01.txt** - название текстового файла задания.

```
[ ] # Открываем файл "24-01.txt" в режиме чтения
with open("/content/24-01.txt", "r") as f:
    # Читаем строку файла в переменную s
    s = f.readline()
    # Записываем искомый символ в переменную k
    k = s[0]
    # Создаем переменную count - сколько раз искомый символ встречается в строке s
    count = 0
    # Создаем цикл для прохождения по всем символам строки s
    for i in s:
        # Проверяем условие: совпадает ли искомый символ k с текущим символом i
        if k == i:
            # Если условие выполняется - значение переменной count увеличивается на 1
            count = count + 1
    # Выводим на экран количество вхождений первого символа k в строку s
    print(count)
```

Алгоритм решения задачи:

- Открываем файл **24-01.txt** и считываем строку файла в переменную **s**.
- Получаем из строки её первый символ **k**, используя срез **s[0]** - это искомый символ, количество которых необходимо подсчитать в строке.
- Создаем переменную **count** со значением **0**, которая будет содержать количество вхождений символа **k** в строку **s**.
- Перебираем все символы строки, используя цикл с параметром **for i in s**, где **s** - искомая строка, **i** - текущий символ строки.
- Сравниваем текущий символ строки с искомым символом **k**: если они совпадают - подсчитываем данный символ, увеличивая значение **count** на **1**.
- Выводим значение переменной **count** на экран.

Диск Доступно: 70.73 GB.

Рис. 12. Пример экзаменационного задания № 1

вый файл, состоящий из одной строки, которая содержит только буквы латинского алфавита, и необходимо определить, сколько раз встречается символ, стоящий в файле на первом месте. В соответствующей кодовой ячейке представлена программа, позволяющая решить данную задачу, а ниже — алгоритм решения задачи с подробными комментариями.

Для того чтобы запустить программу, которая решает задачу, сначала необходимо прикрепить текстовый файл задачи к интерактивному блокноту. Для этого необходимо скачать данный файл (1) на свой компьютер под именем, которое указано в задаче (в данном случае — *24.01.txt*). Затем нужно нажать на кнопку *Файлы* (2) на боковой панели интерактивной облачной среды, в появившейся колонке выбрать кнопку *Загрузить в сессионное хранилище* (3), выбрать искомый текстовый файл и нажать на кнопку *Открыть*. При первой загрузке файла в интерактивный блокнот выводится предупреждение о том, что загруженные файлы будут удалены при перезапуске облачной среды. Загруженный файл *24.01.txt* (4) отобразится в списке файлов ниже папки *sample_data*. При нажатии на файл правой кнопкой мыши появится контекстное меню (5), используя которое можно скачать файл на свой компьютер, переименовать или удалить файл, а также скопировать путь к нему. Поскольку все файлы, загружаемые в интерактивный блокнот, находятся в папке *content*, текстовый файл будет доступен по следующему пути: */content/24.01.txt*. После загрузки файла обучающийся может запустить программу, нажав на кнопку запуска, которая появляется при наведении курсора на символы «[]», и увидеть результат работы программы.

Раздел «Список тренировочных заданий» представлен на рисунке 13. Данный раздел содержит тренировочные задания, направленные на подготовку обучающихся к решению задания № 24 ЕГЭ. Прежде чем приступить к работе, обучающемуся необходимо

войти в свою учетную запись Google и нажать на кнопку *Копировать на диск*, чтобы скопировать интерактивный блокнот на свой Google Диск. Для решения тренировочных заданий учащийся может воспользоваться теоретическим материалом интерактивного блокнота и текстовыми файлами заданий, которые можно скачать на свой компьютер, используя представленные в блокноте ссылки.

Для проверки ответов обучающихся и создания собственных тестовых текстовых файлов к тренировочным заданиям учитель может использовать **интерактивный блокнот «Задание № 24 ЕГЭ по информатике» (для учителя)** (рис. 14):

<https://colab.research.google.com/drive/1mpTM4FCZdF2YprLd7aQYMxPKkuVC6mWm?usp=sharing>.

Данный блокнот состоит из трех разделов:

- «Создание тестовых файлов к тренировочным заданиям»;
- «Ответы к тренировочным заданиям»;
- «Ответы к тренировочным заданиям для скачивания».

Последние два раздела содержат ответы к тренировочным заданиям блокнота «Задание № 24 ЕГЭ по информатике» (для обучающихся) как непосредственно в самом блокноте, так и в форме ссылки на архив с текстовыми файлами и решениями тренировочных заданий в виде программ на языке Python.

В разделе «Создание тестовых файлов к тренировочным заданиям» представлена программа, позволяющая создавать текстовые файлы к заданиям, подобным тем тренировочным заданиям, которые рассматривались в интерактивном блокноте «Задание № 24 ЕГЭ по информатике» (для учителя) (рис. 15). Чтобы сгенерировать текстовый файл, учителю достаточно изменить имя файла в константе *FILENAME*, в переменной *N* ввести количество символов, которые должен

Интерактивный блокнот «Задание №24 ЕГЭ по информатике» (для обучающихся) ☆

Файл Изменить Вид Вставка Среда выполнения Инструменты Справка Изменения сохранены

Комментировать Поделиться

ОЗУ Диск Редактирование

Файлы

+ Код + Текст

Список тренировочных заданий

Инструкция: для закрепления полученных знаний выполните следующие тренировочные задания. Прежде чем приступить к заданиям, нажмите на кнопку "Копировать на диск", чтобы сохранить результаты работы.

№1. Текстовый файл состоит из одной строки, которая содержит только буквы латинского алфавита. Определите, сколько раз в исходном файле встречается символ *N*. Для выполнения этого задания следует написать программу.

Текстовый файл задания доступен по следующей ссылке: <https://drive.google.com/file/d/1zLXa6MDUfOuT-9lpeXCKjJb10tbMpACC/view?usp=sharing>

[] # Напишите ниже программу для тренировочного задания №1 на языке Python.
Для работы с файлом задания, подключите его во вкладке "Файлы".
Ссылка на файл задания: `"/content/24_1.txt"`.

№2. Текстовый файл состоит только из символов *P*, *Q* и *R*. Определите максимальное количество идущих подряд символов, среди которых каждые два соседних различны. Для выполнения этого задания следует написать программу..

Текстовый файл задания доступен по следующей ссылке:
<https://drive.google.com/file/d/13wH07PnT9WEw6NQYvc43G9fbyLFLgOKI/view?usp=sharing>

[] # Напишите ниже программу для тренировочного задания №2 на языке Python.
Для работы с файлом задания, подключите его во вкладке "Файлы".
Ссылка на файл задания: `"/content/24_2.txt"`.

Диск Доступно: 70.73 GB.

Рис. 13. Раздел «Список тренировочных заданий»

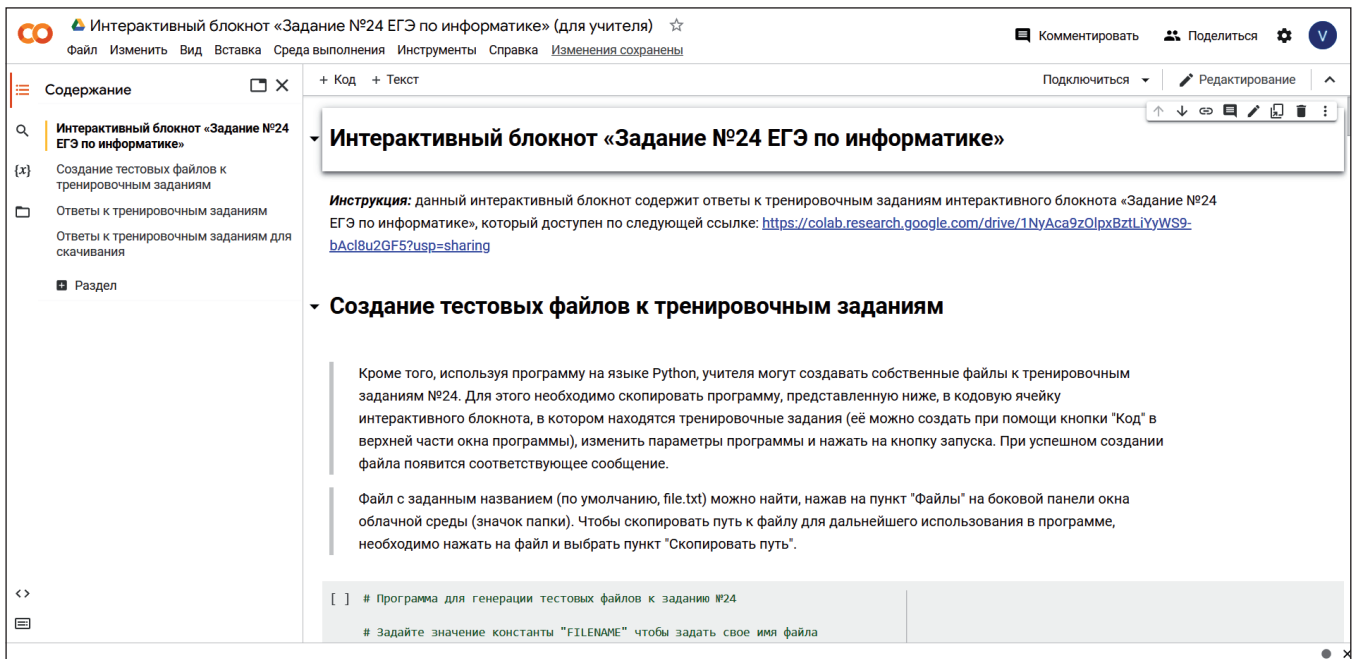


Рис. 14. Интерактивный блокнот «Задание № 24 ЕГЭ по информатике» (для учителя)

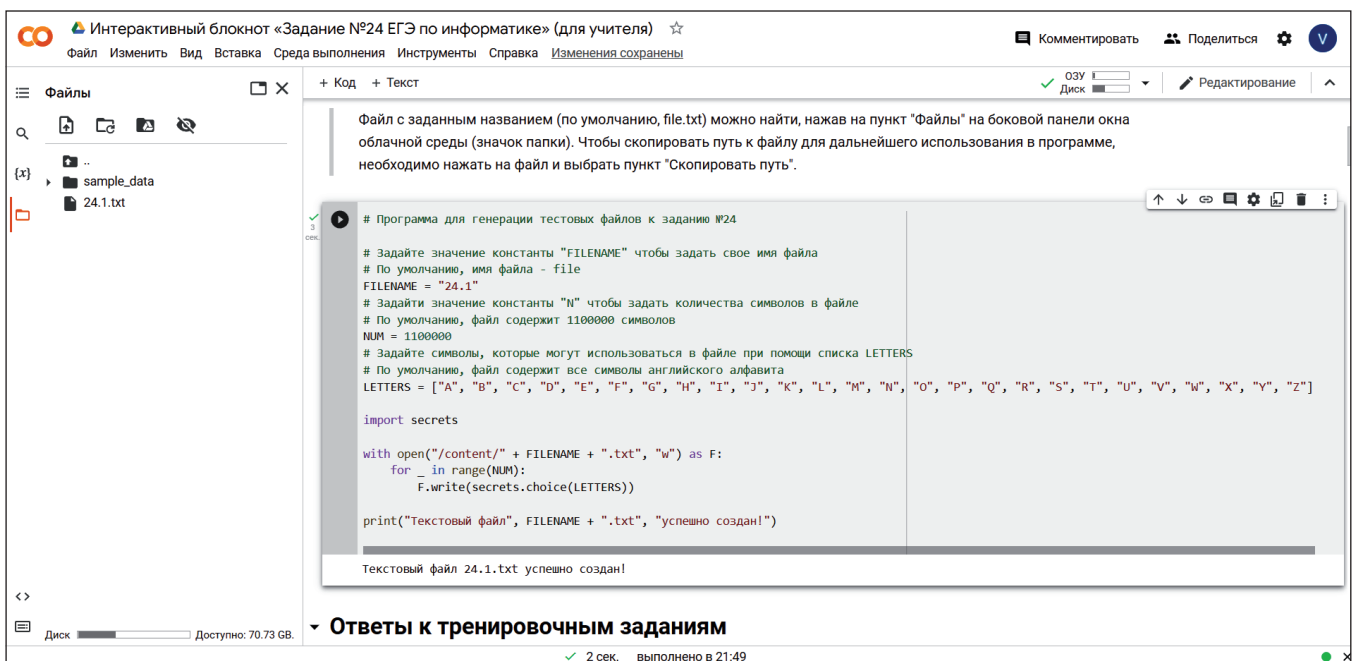


Рис. 15. Программа для генерации файлов к тренировочным заданиям

содержать файл, изменить список символов LETTERS, оставив в нем только те символы, которые должен содержать файл, а затем нажать на кнопку запуска программы. При успешном создании файла в качестве результата программа выведет сообщение «Текстовый файл успешно создан», а в колонке *Файлы* появится файл с заданным названием, который можно скачать, используя контекстное меню.

Кроме того, текст данной программы учитель может скопировать в собственный интерактивный блокнот, чтобы сгенерированные файлы могли использоваться для решения задач в облачной среде, без необходимости скачивания файлов на компьютер.

4. Заключение

Интерактивная облачная среда Google Colaboratory, с одной стороны, позволяет учителям использовать готовые обучающие интерактивные блокноты в рамках обучения программированию на базе языка Python в школьном курсе информатики — как на уроках открытия новых знаний, обретения новых умений и навыков, так и на уроках комплексного применения знаний и умений, для демонстрации теоретического материала по соответствующим темам раздела, а также для создания практических работ с целью формирования у обучающихся навыков программирования.

С другой стороны, возможности облачной среды Google Colaboratory позволяют учителям использовать интерактивные блокноты в качестве инструмента для подготовки обучающихся к заданиям по программированию ЕГЭ по информатике.

Список источников

1. *Андреева Е. В.* Задачи по программированию в ЕГЭ по информатике // Информатика и образование. 2015. № 3. С. 54–56. EDN: TQMHDR.
2. *Артюхина М. С.* Особенности современных средств обучения в контексте интерактивных технологий // Вестник Российского университета дружбы народов. Серия: Информатизация образования. 2014. № 2. С. 76–81. EDN: SCNMVX.
3. *Балабанова И. А.* Программирование в «облаке» // Актуальные проблемы методики обучения информатике и математике в современной школе. Материалы международной научно-практической интернет-конференции (Москва, 19–25 апреля 2021 года). М.: Московский педагогический государственный университет, 2021. С. 26–29. EDN: XSMPEQ.
4. *Босова Л. Л.* Как учат программированию в XXI веке: отечественный и зарубежный опыт обучения программированию в школе // Информатика в школе. 2018. № 6. С. 3–11. EDN: XZOOJV.
5. *Газейкина А. И., Тупицына М. В.* Методика формирования у учащихся основной школы умения осуществлять учебное сотрудничество средствами облачных технологий // Педагогическое образование в России. 2017. № 6. С. 21–30. DOI: 10.26170/ro17-06-03. EDN: ZDLZWD.
6. *Голицына И. Н., Афзалова А. Н.* Использование облачных вычислений в образовательном процессе // Образовательные технологии и общество. 2014. № 2. С. 450–459. EDN: SEFPMZ.
7. *Касьянов С. Н.* Обучение языку программирования Python в онлайн-сообществе учащихся школ // Известия Волгоградского государственного педагогического университета. 2021. № 6. С. 50–54. EDN: FGYVWR.
8. *Маркелов В. К., Завьялова О. А.* Возможности языка Python 3.10 при обучении программированию в школе // Шуйская сессия студентов, аспирантов, педагогов, молодых ученых. Материалы XIV Международной научной конференции, посвященной Году науки и технологий Российской Федерации, 205-летию начала подготовки педагогов в Ивановской области (Москва — Иваново — Шуя, 06–07 октября 2021 года). Шуя: Шуйский филиал федерального государственного бюджетного образовательного учреждения высшего образования «Ивановский государственный университет», 2021. С. 96–97. EDN: XXZKLZ.
9. *Митин А. Н.* Облачные технологии в образовании // Вестник НГИЭИ. 2016. № 8. С. 41–47. EDN: WJDAPF.
10. *Фролов Ю. В., Сахнюк П. А.* Цифровые технологии и цифровые компетенции в образовательном процессе подготовки бизнес-информатиков // Актуальные проблемы теории и практики обучения математике, информатике и физике в современном образовательном пространстве. Курск: Курский государственный университет, 2018. С. 60–63. EDN: ZBUWQH.