

УДК 37.04:371:372.862

DOI

ОБУЧЕНИЕ АНАЛИЗУ ЗНАЧЕНИЙ АРИФМЕТИЧЕСКИХ ВЫРАЖЕНИЙ С ИСПОЛЬЗОВАНИЕМ ЯЗЫКА ПРОГРАММИРОВАНИЯ PYTHON В ШКОЛЕ

¹Козлов С.В., ²Быков А.А.

¹ФГБОУ ВО «Смоленский государственный университет», Смоленск,
e-mail: svkozlov1981@yandex.ru;

²ФГБОУ ВО «Национальный исследовательский университет «МЭИ»,
Смоленский филиал, Смоленск, e-mail: alex1by@mail.ru

Дидактическая линия изучения систем счисления составляет основу школьного курса информатики. Многие задачи прикладных вычислений эффективно решаются с помощью ее понятийного аппарата. При этом алгоритмы выполнения заданий многих смежных тем невозможно построить без знаний правил работы в разных системах счисления и умений оперировать их инструментами. В то же время при решении таких задач целесообразно использовать вычислительные возможности современных сред программирования. В связи с этим при обучении анализу значений арифметических выражений в школе актуально применять инструменты языка программирования Python. Ввиду этого в статье обсуждается обращение к инструментам программной среды PyCharm при оперировании значениями громоздких арифметических выражений. В частности, акцент делается на анализе значений арифметических выражений при их переводе в разные системы счисления из десятичной записи. Авторами обсуждаются вопросы решения таких задач в контексте профильной подготовки обучаемых по информатике. В разрезе подготовки к экзамену по профильной информатике в школе рассматриваются особенности решения задания единого государственного экзамена по анализу значения арифметического выражения с использованием языка Python. Актуальность педагогического исследования обусловлена необходимостью внедрения в образовательный процесс современных подходов программирования для решения практических задач.

Ключевые слова: информатика, обучение, алгоритм, язык программирования, Python, системы счисления, вычисления, числовые выражения, образовательный процесс

TEACHING TO ANALYZE THE VALUES OF ARITHMETIC EXPRESSIONS USING PYTHON PROGRAMMING LANGUAGE AT SCHOOL

¹Kozlov S.V., ²Bykov A.A.

¹Smolensk State University, Smolensk, e-mail: svkozlov1981@yandex.ru;

²National Research University "MPEI", Smolensk branch, Smolensk, e-mail: alex1by@mail.ru

The didactic line of studying number systems forms the basis of the school course in computer science. Many problems of applied computing are effectively solved using its conceptual apparatus. At the same time, algorithms for completing tasks on many related topics cannot be constructed without knowledge of the rules of working in different number systems and the ability to operate their tools. At the same time, when solving such problems, it is advisable to use the computing capabilities of modern programming environments. In this regard, when teaching how to analyze the results of arithmetic expressions at school, it is important to use the tools of the Python programming language. In view of this, the article discusses turning to the tools of the PyCharm software environment when operating with the values of cumbersome arithmetic expressions. In particular, the emphasis is on analyzing the meanings of arithmetic expressions when converting them into different number systems from decimal notation. The authors discuss the issues of solving such problems in the context of specialized training of students in computer science. In terms of preparing for the exam in specialized computer science at school, the features of solving the Unified State Examination task on analyzing the value of an arithmetic expression using the Python language are considered. The relevance of pedagogical research is due to the need to introduce modern programming approaches into the educational process to solve practical problems.

Keywords: computer science, training, algorithm, programming language, Python, number systems, calculations, numerical expressions, educational process

Знакомство с понятиями систем счисления и изучение алгоритмов преобразования целых чисел в них составляют одну из базовых дидактических линий курса информатики в средней школе [1, 2]. Выполнение арифметических действий в разных системах счисления относится к умениям, которые получают обучающиеся на уроках профильной информатики. Другую фун-

даментальную дидактическую линию школьной информатики составляет курс программирования [3, 4]. В нем изучают основные типы данных, методы их обработки, базовые алгоритмические структуры, фундаментальные подходы к построению программ, приемы и способы разработки компьютерных программ. В то же время навыки, полученные при работе в средах про-

граммирования, нередко вообще не рассматриваются на уроках при изучении других тем как эффективный инструмент решения поставленных задач.

Так, при решении задач анализа значений арифметических выражений многие учителя в старших классах школы зачастую не используют возможности современных систем программирования, например таких, как инструментальная среда языка Python [5]. Они отводят в 10–11 классах все время по теме «Системы счисления» на получение умений и отработку навыков выполнения арифметических действий с большими числами. На уроках изучают общие приемы и правила оперирования цифрами числовых выражений в разных системах счисления. При этом все вычисления проводятся «вручную», упор по-прежнему делается на теоретические аспекты преобразования арифметических выражений. Применение базовых алгоритмов обработки больших массивов данных не рассматривается. Многие учителя не демонстрируют связь теории и практики, не показывают единую концепцию тем и разделов курса информатики в школе.

Это становится очевидным при проверке знаний анализа и преобразования арифметических выражений на едином государственном экзамене (ЕГЭ) по информатике [6]. Обучающиеся нерационально тратят время при решении задания № 15, в котором при записи арифметического выражения показатели степени представляют собой четырехзначные числа. Они, как правило, выбирают промежуточный вариант решения такой задачи. На черновике расписывают арифметические действия «столбиком», после чего вычисление требуемого количества цифр в значении выражения определяют с помощью программного приложения «Калькулятор». Это приводит при знании правильной последовательности действий для решения задания к множественным неоправданным ошибкам вычислительного характера, которых можно было бы избежать. Использование программного подхода при составлении алгоритма на языке Python позволяет «свести к нулю» вычислительные ошибки, оптимизировать алгоритм решения с помощью компьютерных средств обработки больших данных [7, 8].

Цель исследования – оценка эффективности объяснения обучающимся способов построения и выполнения алгоритмов для вычисления значений простейших выражений в разных системах счисления с использованием языка программирования Python.

Научная новизна состоит в методологическом использовании конструкций и функций языка программирования Python

для решения практических заданий при анализе значений арифметических выражений, представленных в разных системах счисления.

Материалы и методы исследования

Изучение языка программирования Python на базовом уровне относится к 8–9 классам средней школы. Затем оно продолжается в профильных классах старшей школы. В то же время применить навыки программирования на практике обучающимся достаточно сложно. В основном он решает классические, устоявшиеся со временем задачи на обработку целых и вещественных чисел, логических и строковых данных. Безусловно, без знания подходов к решению таких заданий профессиональный кругозор обучающихся при изучении информатики на профильном уровне был бы неполным.

При этом не менее важно приложение всех своих алгоритмических навыков при анализе заданий по обработке информации из разных сфер практической деятельности человека [9]. Например, при вычислении больших или малых значений, которые свойственны биржевым котировкам при прогнозировании реакций на поведение игроков на рынке и составлении трендов событий. Отметим, что задание № 14 компьютерного ЕГЭ по информатике носит именно такой прикладной характер. В этом задании обучающимся предоставляется для анализа «громоздкое» арифметическое выражение, значение которого требуется вычислить и определить количество цифр в нем в заданной системе счисления. Подобные задания прикладного характера можно встретить и при решении других школьных заданий, некоторые из них также представлены в списке заданий ЕГЭ по информатике. Например, в задании № 25 при проверке делимости больших целых чисел или выборе больших числовых значений, которые удовлетворяют заданной маске. В связи с этим такая целенаправленная подготовка обучающихся к решению заданий прикладного характера с привлечением инструментов систем программирования требует особого подхода при организации учебного процесса по информатике.

Использование языка программирования Python позволяет оформить решение таких заданий в виде небольших программ [10, 11]. Число строк в них относительно невелико, оно колеблется от 8 до 10. При этом интерпретация используемых в алгоритме команд на естественный язык задачи не вызывает у обучающихся дополнительных затруднений. Действия, определяемые в программе, известны им по классическим

приемам составления компьютерных приложений. От учеников требуется воспользоваться известным алгоритмом и применить в нем вспомогательные конструкции, которые позволяют определить параметры промежуточных вычислений.

Заметим, что выбранная инструментальная среда языка программирования Python не накладывает знание обучающимися дополнительного материала. Решение строится на использовании композиции базовых алгоритмических конструкций. В цикле используется конструкция неполного ветвления, или условный оператор применяется после его выполнения. В таком случае применяется одна из встроенных функций для вывода ответа на вопрос задачи о характеристике полученного результата. Также отметим, что алгоритм вычислений целесообразно оформить в виде отдельной функции, так как в ряде заданий его требуется применить неоднократно для различных входных значений, как, например, в задании № 5 ЕГЭ по информатике.

Проанализируем решение такой задачи, аналогичной заданию № 14 из компьютерного ЕГЭ по информатике.

Пример. Вычислите, сколько цифр 6 содержит значение арифметического выражения $7^{2024} - 4 \cdot 7^{835} + 3 \cdot 7^{126} - 17$ в системе счисления с основанием 7.

Рассмотрим алгоритм решения данной задачи, который может быть реализован в программной среде PyCharm средствами языка Python.

```
def f(x):
    k = 0
    while x > 0:
        c = x % 7
        if c == 6:
            k += 1
        x = x // 7
    return k
x = 7 ** 2024 - 4 * 7 ** 835 + 3 * 7 ** 126 - 17
print(f(x))
```

Искомый ответом на вопрос задачи будет число 1312. Отметим отдельные особенности приведенного алгоритма решения задачи. Для вычисления количества цифр 6 в записи арифметического выражения, представленного в семеричной системе счисления, в программе была реализована отдельная пользовательская функция. В основу ее работы заложен алгоритм перевода натурального числа в семеричную систему счисления. В цикле с предусловием while на каждом его витке выполняется выделение очередной цифры семеричной записи числа x . После выделения каждой цифры проводится проверка, является ли она циф-

рой 6. Если да, то счетчик k таких цифр увеличивается на единицу. Предварительно до выполнения цикла начальное значение данного счетчика было задано равным нулю. Когда текущая цифра обработана, исходное число x уменьшается в 7 раз с помощью операции целочисленного деления, то есть текущая выделенная цифра в семеричной записи числа удаляется из дальнейшего рассмотрения. Цикл завершается в том случае, когда число x станет равным нулю, то есть в нем не останется цифр для перевода в семеричную запись.

Собственно тело программы состоит всего из двух строк. В первой из них с помощью арифметических операций и операции возведения в натуральную степень задается значение выражения для перевода в семеричную систему счисления. Во второй строке с помощью команды вывода значений на экран консоли print показано искомое число цифр 6 в семеричной записи обрабатываемого арифметического выражения. Если в другой аналогичной задаче требуется перевести значение арифметического выражения в систему счисления n и определить, сколько раз встречается в этой записи цифра m , то в цикле while выделение остатка от деления и результат целочисленного деления осуществляется на n , а цифра в операторе if сравнивается с m .

Отметим важные особенности такого подхода к решению задачи. Во-первых, на примере данного задания обучающимся можно продемонстрировать отличие написания функций в языке программирования Python. Если в описанной функции использовать оператор return, то алгоритм будет возвращать вычисленное значение в основную программу. Иными словами, функция будет работать как функция. Если же всего заменить команду return k на команду print(k), то пользовательская функция не будет возвращать значений в основную программу. Она будет работать как процедура. Она будет выполнять тот же самый определенный в ее алгоритме набор инструкций и будет выводить вычисленное значение числа цифр 6 в семеричной записи заданного арифметического выражения. При этом в программе также будет необходимо изменить вывод вычисленного значения в команде print на вызов пользовательской функции инструкцией f(x).

Во-вторых, обучающимся можно продемонстрировать использование в программном коде встроенных функций обработки символьных данных. При обычной практике преподавания в школьном курсе информатики упор, как правило, делается на обработке числовых данных. Для этого

в алгоритме необходимо сформировать семичисленную запись выражения в текстовой форме. Для этого следует предусмотреть символьную переменную s , присвоить ей до выполнения цикла начальное значение в виде пустой строки. А затем на каждой витке цикла приписывать к ее записи слева очередную выделяемую цифру. При этом необходимо помнить, что цифры выделяются в алгоритме в обратном порядке, поэтому их приписывают именно слева к символьной записи числа. В противном случае запись придется после цикла еще перевернуть. Это также можно продемонстрировать обучающимся как один из способов работы со срезами строк в языке программирования Python, с помощью команды $s[::-1]$, которая разворачивает строку в обратном порядке.

В-третьих, алгоритм перевода в семичисленную систему счисления можно унифицировать для любого основания. Для этого необходимо определить функцию f от двух параметров. Первый параметр, как и был, будет определять обрабатываемое значение арифметического выражения. Второй параметр будет представлять собой основание системы счисления, в которую требуется перевести значение выражения. При таком подходе можно получить универсальную функцию перевода заданного арифметического выражения в указанную систему счисления с определением числа заданных цифр в ней. Это обеспечит расширение возможностей использования пользовательской функции в приложениях с массовой обработкой числовых данных для разных систем счисления.

Данный способ решения задачи № 14 из компьютерного ЕГЭ по информатике, а также аналогичных практических заданий раскрывает более глубокие возможности современных инструментальных сред. Ввиду этого обучающиеся могут получить прочные знания по фундаментальным основам построения компьютерных программ. Это не только обеспечивает высокий уровень выполнения задания № 14 ЕГЭ по информатике, но и является инструментом качественного выполнения других заданий экзамена, связанных с их решением с использованием систем программирования. Кроме того, такой подход снижает вычислительную сложность предлагаемого задания. Количество ошибок, допускаемых при «ручном» подсчете искомого цифр в значении арифметического выражения, уменьшается.

Подобные задания предлагались обучающимся в ходе педагогического эксперимента по теме «Анализ и вычисление значений арифметических выражений с помощью инструментов языка Python». В сре-

де PyCharm они приобретали и закрепляли навыки практического выполнения таких задач, анализировали разные методы их решения. Методология экспериментальной деятельности обуславливала проведение констатирующего и формирующего педагогического исследования. Для анализа результатов работы были использованы математические методы обработки статистических данных. Гипотеза исследования состояла в том, что использование возможностей языка Python и среды PyCharm в школе при вычислении и анализе значений арифметических выражений повышает эффективность обучения.

Экспериментальная педагогическая деятельность по изучению методов составления и анализа алгоритмов для вычисления значений арифметических выражений с помощью языка программирования Python осуществлялась в двух образовательных учреждениях г. Смоленска Смоленской области. Одной экспериментальной площадкой был Смоленский физико-математический лицей при МИФИ, другой – средняя школа № 6 г. Смоленска. Педагогический эксперимент проводился в двух профильных классах, физико-математическом классе и IT-классе соответственно. В эксперименте участвовали 32 обучающихся.

Результаты исследования и их обсуждение

На констатирующем этапе педагогического эксперимента обучающиеся выполняли задания на анализ значений арифметических выражений в тетради. Они изучали, как определить положение значащей цифры в числе в заданной системе счисления. Для этого они преобразовывали исходное выражение в виде суммы одночленов, каждый из которых должен был представлять собой произведение коэффициента на основание системы счисления в некоторой степени n . Положение коэффициента определяло его позицию $(n - 1)$ в числе в заданной системе счисления. Затем с учетом знаков в выражении перед такими одночленами формировались уменьшаемое и вычитаемое, которые соответствовали знакам плюс и минус. Они записывались друг под другом, после чего требовалось произвести вычитание столбиком. В полученной разности вычислялось количество указанных в задании цифр. Для этого обучающиеся пользовались формулой подсчета числа целых чисел на отрезке $[a; b]$, а именно $b - a + 1$. На заключительном занятии обучающимся был предложен тест. Диагностика включала семь тестовых заданий. Все тестовые задания было необходимо выполнить в тетради.

На формирующем этапе педагогического эксперимента обучающиеся выполняли аналогичные практические задания. При этом они использовали функциональные средства в приложении PyCharm. Обучающиеся изучали правила записи арифметических выражений в его программной среде и изучали алгоритм перевода десятичного числа в указанную систему счисления. При этом акцент делался как на умении формировать собственно запись числа в указанной при переводе системе счисления, так и оперировать ее отдельными цифрами. Параллельно обучающиеся овладевали навыками записи пользовательских функций, они также изучали отличие в них просто выполнения набора действий, от вычисления значений и возвращения их в основную программу с помощью команды return. В завершение этого этапа эксперимента с обучающимися было проведено тестирование. На итоговом занятии им предстояло решить также семь заданий. В то же время вопросы в заданиях носили более разнообразный характер. Кроме того, они характеризовались повышенным уровнем сложности. Например, в одних из них требовалось вычислить сумму цифр в записи заданного арифметического выражения в указанной системе счисления. В других – определить, какая цифра в ней встречается реже или чаще.

Тестирование на каждом из этапов проводилось с использованием возможностей автоматизированной программной оболочки «Advanced Tester» [12, 13]. Она позволяет предлагать обучающимся как индивидуальные, так и групповые тесты. Индивидуальные тесты использовались ими

при изучении учебного материала, как непосредственно на уроках, так и в онлайн-режиме [14, 15]. Групповые тесты завершали каждый из этапов педагогического эксперимента. Данные диагностик знаний и умений обучающихся обрабатывались автоматически. Для этого были задействованы ресурсы программной среды «Advanced Tester». В этой программе на основании интеллектуальных методов обработки данных с использованием соответствия Галуа и импликативных матриц обобщались результаты экспериментальной работы. Они сведены в две нижеследующие табл. 1 и 2.

*Качественный анализ условий
и результатов эксперимента*

Экспериментальная работа и анализ полученных в ее ходе данных доказывает гипотезу исследования. Расширение применения функционала среды PyCharm при вычислении значений арифметических выражений отражает рост знаний и умений обучающихся. Учеников, остановившихся в познании изучаемого материала на базовом уровне, практически не остается – всего два человека. Обучающиеся формируют группы с углубленными предметными навыками. Эти факты позволяют констатировать факт упрощения исследования операндов и действий над ними при вычислении значений арифметических выражений. Данная ситуация обусловлена тем, что преобразование громоздких выражений с многозначными показателями степени требует при «ручном» варианте решения значительной концентрации внимания и умения тонко оперировать большими числами.

Таблица 1

Результаты констатирующего этапа педагогического эксперимента

Группа	Число обучающихся, достигших уровня усвоения знаний			Всего
	Высокий	Повышенный	Базовый	
СФМЛ при МИФИ	4	6	8	18
IT-класс школа № 6	1	6	7	14
Всего	5	12	15	32

Таблица 2

Результаты формирующего этапа педагогического эксперимента

Группа	Число обучающихся, достигших уровня усвоения знаний			Всего
	Высокий	Повышенный	Базовый	
СФМЛ при МИФИ	8	9	1	18
IT-класс школа № 6	9	4	1	14
Всего	17	13	2	32

Использование компьютерных возможностей смещает акцент на запись алгоритма вычислений, несомненно, при правильной интерпретации в систему его арифметических действий над заданными операндами. Кроме того, использование языка Python существенно расширяет круг решаемых задач на вычисление значений арифметических выражений, что выражается, например, в наборе вопросов к исследуемым на практике ситуациям. Наряду с этим время на их анализ сокращается, тем самым высвобождается учебный ресурс на более детальное и глубокое изучение данной тематики. Таким образом, у обучающихся закладываются прочные основы для составления алгоритмов программ на материале решения прикладных задач одним из наиболее эффективных способов.

Заключение

Итак, результаты работы, полученные в экспериментальном исследовании, указывают на то, что применение функциональных средств языка Python полезно при рационализации решения задач на вычисление значений арифметических выражений. Обучающиеся, выполняя такие задания, получают не только навыки составления алгоритмов и их программной реализации, они осваивают специфику записи функций в языке Python. Это позволяет сформировать у них фундаментальную базу знаний по профильной информатике, основу которой составляет применение функций для организации подпрограмм при записи алгоритмов. Обучающиеся начинают осознанно выбирать алгоритмический способ записи функций в соответствии с поставленной практической задачей. Таким образом, область приложения профессиональных умений обучающихся расширяется, что позволяет сформировать грамотного IT-специалиста.

Список литературы

1. Латышева Е.В. Тема «Системы счисления» в школьном курсе информатики // Информационные технологии в образовательном процессе вуза и школы. Материалы XV Всероссийской научно-практической конференции. Редколлегия: Р.М. Чудинский (науч. ред.) и др. Воронеж, 2021. С. 269–273.
2. Иорданский М.А., Мухин Н.А. Использование информационно-педагогических технологий при изучении систем счисления и компьютерной арифметики // Информатика и образование. 2017. № 9 (288). С. 51–53.
3. Бровка Н.В., Францкевич А.А. Обучение учащихся основам алгоритмизации и программирования // Педагогика информатики. 2020. № 3. С. 1–9.
4. Козлов С.В. Особенности обучения школьников информатике в профильной школе // Концепт. 2014. № 1. С. 31–35.
5. Пономарев Д.А. Решение олимпиадных задач по информатике по теме «Системы счисления» на языке программирования Python // Информационные технологии в образовании. 2021. № 4. С. 193–198.
6. Старикова М.Е. Системы счисления в ЕГЭ по информатике // Информатика: проблемы, методология, технологии: сборник материалов XIX международной научно-методической конференции / Под ред. Д.Н. Борисова. Воронеж, 2019. С. 1846–1848.
7. Каган Э.М. Применение визуальных языков программирования для повышения эффективности обучения разделу «Алгоритмизация и программирование» школьного курса информатики // Вестник Московского городского педагогического университета. 2018. № 1 (43). С. 99–104.
8. Козлов С.В., Быков А.А. Обучение школьников выполнению и анализу алгоритмов для формального исполнителя с использованием систем программирования // Современные проблемы науки и образования. 2023. № 2. URL: <https://science-education.ru/ru/article/view?id=32491> (дата обращения: 15.03.2024). DOI: 10.17513/spno.32491.
9. Архангельская Е.В. Организация обучения основам алгоритмизации и программирования с использованием анализа математических задач // Информатизация образования и науки. 2022. № 4 (56). С. 166–175.
10. Рослякова Е.А., Химич А.М. Преимущества использования языка программирования Python при изучении раздела «Алгоритмизация и программирование» в школьном курсе информатики // Современные тенденции развития фундаментальных и прикладных наук: материалы Всероссийской с международным участием научно-практической конференции / Под ред. С.А. Коньшаковой. 2018. С. 148–152.
11. Ильченко О.Ю., Сырицына В.Н., Кадеева О.Е. Решение задач ЕГЭ по информатике средствами языка Python // Высшее образование сегодня. 2021. № 11–12. С. 42–54.
12. Козлов С.В., Быков А.А. Особенности изучения междисциплинарных тем школьных курсов математики и информатики с помощью методов математического моделирования // Проблемы современного образования. 2021. № 5. С. 250–261.
13. Козлов С.В., Быков А.А. Применение методов математического моделирования для диагностики знаний школьников // Современные наукоемкие технологии. 2021. № 4. С. 157–162. DOI: 10.17513/snt.38632.
14. Киселева О.М. Программные средства поддержки удаленного обучения // Вызовы цифровой экономики: тренды развития в условиях последствий пандемии COVID-19: сборник статей IV Всероссийской научно-практической конференции, приуроченной к Году науки и технологий в России. Брянск, 2021. С. 143–146.
15. Senkina G.E., Timofeeva N.M., Kiseleva O.M. Modernization of traditional educational forms in the context of distance learning // Journal of Higher Education Theory and Practice. 2022. Т. 22, № 3. Р. 160–165.