

**Попов Владислав Сергеевич**

Старший преподаватель кафедры
«Информационные системы и телекоммуникации»
МГТУ им. Н.Э. Баумана

По следам летней школы для учителей информатики 2023 в МГУ им. М.В. Ломоносова: способы решения задания на системы счисления № 14 в ЕГЭ

Демонстрационный вариант контрольных измерительных материалов единого государственного экзамена 2024 года по информатике включает сразу два типа заданий № 14 по теме системы счисления (рис. 1). В данной статье приведены решения этих заданий на языке программирования Python, под-

14

Операнды арифметического выражения записаны в системе счисления с основанием 19.

$$98897x21_{19} + 2x923_{19}$$

В записи чисел переменной x обозначена неизвестная цифра из алфавита 19-ричной системы счисления. Определите наибольшее значение x , при котором значение данного арифметического выражения кратно 18. Для найденного x вычислите частное от деления значения арифметического выражения на 18 и укажите его в ответе в десятичной системе счисления. Основание системы счисления указывать не нужно.

Ответ: _____.

ИЛИ

Значение арифметического выражения

$$3 \cdot 3125^8 + 2 \cdot 625^7 - 4 \cdot 625^6 + 3 \cdot 125^5 - 2 \cdot 25^4 - 2024$$

записали в системе счисления с основанием 25. Сколько значащих нулей содержится в этой записи?

Ответ: _____.

робное описание предлагаемых программ и ссылки на более ранние публикации по теме.

Союз «или» обозначает, что выпускник может встретить как первый, так и второй тип задания на экзамене в собственном КИМе. Соответственно, нужно быть готовым к решению и проверке сразу двух типов заданий.

Тип 1

Формулировка задания:

Операнды арифметического выражения записаны в системе счисления с основанием 19.

$$98897x21_{19} + 2x923_{19}$$

В записи чисел переменной x обозначена неизвестная цифра из алфавита 19-ричной системы счисления. Определите наибольшее значение x ,

при котором значение данного арифметического выражения кратно 18. Для найденного x вычислите частное от деления значения арифметического выражения на 18 и укажите его в ответе в десятичной системе счисления. Основание системы счисления указывать не нужно.

Решение задания в Python (использование функции int):

```
digits = "0123456789ABCDEFGHI"
print("x quotient")
for x in digits:
    s = int("98897" + x + "21", 19) + int("2" + x + "923", 19)
    if s % 18 == 0:
        print(x, s // 18)
```

Вывод программы:

```
x quotient
6 469030538
F 469034148
```

Решение задания в Python (полиномиальное представление чисел):

```
print("x quotient")
for x in range(19):
    s = (9*19**7 + 8*19**6 + 8*19**5 + 9*19**4 + 7*19**3 +
x*19**2 + 2*19 + 1) + (2*19**4 + x*19**3 + 9*19**2 + 2*19 + 3)
    if s % 18 == 0:
        print(x, s // 18)
```

Вывод программы:

```
x quotient
6 469030538
15 469034148
```

Ответ: 469034148

В первой строке первого варианта решения в переменную `digits` записаны цифры 19-ричной системы счисления: 10 цифр десятичной системы счисления от 0 до 9 и 9 первых букв английского алфавита от A до I. Проблема при формировании строки возможных цифр 19-ричной системы счисления заключается в том, что ученики зачастую не помнят английский алфавит, количество и порядок букв в нём. Помочь в решении этой проблемы может знание о том, что максимальная цифра любой позиционной системы счисления на единицу меньше основания данной системы счисления, и функция `int`, осуществляющая перевод заданной первым аргументом строки из заданной вторым аргументом системы счисления в десятичную систему счисления. Если буква I выступает в роли последней цифры 19-ричной системы счисления, то команда `print(int("I", 19))` выведет значение 18 по аналогии с тем, как для 16-ричной системы счисления команда `print(int("F", 16))` выведет значение 15. Таким образом, буквы, используемые в заданной системе счисления, можно подобрать на экзамене с помощью функции `int`.

Во второй строке программы выведена шапка для простоты восприятия последующей таблицы: `print("x quotient")`. Слово «quotient» переводится с английского языка как «частное, отношение, результат деления».

В третьей строке программы с помощью цикла `for` производится перебор всех значений переменной `x`, в которую последовательно присваиваются символы-цифры из строки

`digits`. Говорят, что цикл `for x in digits` осуществляет итерирование по строке.

В теле цикла 19-ричные числа `98897x2119` и `2x92319` с помощью функции `int` преобразуются в десятичную систему счисления и суммируются:

$$s = \text{int}("98897" + x + "21", 19) + \text{int}("2" + x + "923", 19).$$

Первым аргументом функций `int` служат строки, представляющие записи чисел в 19-ричной системе счисления, второй аргумент функций `int`, равный 19, является основанием системы счисления, из которой осуществляется перевод в десятичную систему счисления. Необходимо заметить, что максимальным основанием системы счисления для функции `int` является 36. При основании системы счисления больше 36 необходимо представлять заданные числа как полиномы для их перевода в десятичную систему счисления:

$$s = (9 \cdot 19^{**7} + 8 \cdot 19^{**6} + 8 \cdot 19^{**5} + 9 \cdot 19^{**4} + 7 \cdot 19^{**3} + x \cdot 19^{**2} + 2 \cdot 19 + 1) + (2 \cdot 19^{**4} + x \cdot 19^{**3} + 9 \cdot 19^{**2} + 2 \cdot 19 + 3),$$

при этом значения переменной `x` нужно перебирать в виде: `for x in range(19)`, где 19 – основание системы счисления, в данном случае в переменную `x` присваиваются не 19-ричные цифры, а их десятичные представления – десятичные числа от 0 до 18.

В случае если значение полученной суммы кратно 18, происходит

вывод значения переменной x и искомого частного от деления значения арифметического выражения на 18:

```
if s % 18 == 0: print(x, s // 18).
```

В данном задании требовалось найти частное для наибольшего значения x , поэтому в качестве ответа необходимо выбрать последнее подходящее частное.

Примеры других заданий данного типа, а также решение задания в Excel можно увидеть в статье (Попов В.С., Решаем новую задачу ЕГЭ № 14 в Python и электронной таблицей. Функция int и полиномиальное представление чисел // Потенциал: Математика, Физика, Информатика, № 2 (218), 2023).

Тип 2

Формулировка задания:

Значение арифметического выражения

$$3 \cdot 3125^8 + 2 \cdot 625^7 - 4 \cdot 625^6 + 3 \cdot 125^5 - 2 \cdot 25^4 - 2024$$

записали в системе счисления с основанием 25. Сколько значащих нулей содержится в этой записи?

Решение задания в Python:

```
number = 3*3125**8 + 2*625**7 - 4*625**6 + 3*125**5 - 2*25**4 - 2024
cnt = 0
while number > 0:
    if number % 25 == 0:
        cnt += 1
    number = number // 25
print(cnt)
```

Ответ: 9

Благодаря встроенной длинной арифметике в языке программирования Python возможна запись и хранение очень больших чисел, как это демонстрируется в первой строке программы:

```
number = 3*3125**8 +
+ 2*625**7 - 4*625**6 +
+ 3*125**5 - 2*25**4 - 2024.
```

Во второй строке программы создаётся и инициализируется нулём счётчик количества искоемых цифр: $cnt = 0$.

Цифры числа $number$ в 25-ричной системе счисления извлекаются в цикле `while`, продолжающем итерации, пока значение переменной $number$ больше нуля. Если очередная цифра числа $number$ в 25-ричной системе счисления равна нулю, значение счётчика cnt инкрементируется:

```
if number % 25 == 0: cnt += 1.
```

После условного оператора значение переменной $number$ целочисленно делится на 25, чтобы перейти к следующей цифре числа $number$ в 25-ричной системе счисления.

Возможен вывод десятичных представлений всех цифр числа:

```
number = 3*3125**8 + 2*625**7 - 4*625**6 + 3*125**5 -
2*25**4 - 2024
while number > 0:
    print(number % 25)
    number = number // 25
```

Вывод программы:

```
1 19 21 24 22 24 24 14 0 0 0 0 21 24 1 0 0 0 0 0 3
```

В случае вывода десятичных представлений всех цифр числа ответ на задание следует определить

вручную – для решения представленного задания нужно подсчитать количество нулей.

Для задачи подсчёта количества различных цифр числа удобным решением является использование множества – структуры данных, сохраняющей каждое значение только один раз:

```
number = 3*3125**8 + 2*625**7 - 4*625**6 + 3*125**5 -
2*25**4 - 2024
s = set()
while number > 0:
    s.add(number % 25)
    number = number // 25
print("В 25-ричном числе", len(s), "различных цифр:", s)
```

Вывод программы:

```
В 25-ричном числе 8 различных цифр: {0, 1, 3, 14, 19, 21, 22, 24}
```

Здесь мощность множества s получена с помощью функции `len`. Десятичные представления 25-ричных цифр числа `number` выведены в фигурных скобках:

`{0, 1, 3, 14, 19, 21, 22, 24}`.

Для создания пустого множества использован конструктор `set()` без параметров, для добавления элемента в множество используется метод `add()`.

Подробнее о способе решения заданий данного типа см. (Попов В.С., Решаем задачу ЕГЭ № 14 в семь строк кода на Python // Потенциал: Математика, Физика, Информатика, № 11 (191), 2020).

Рассмотренные компьютерные способы решения задания № 14 ЕГЭ по информатике являются небольшими и лёгкими для запоминания программами, могут быть модифицированы под задачу и позволяют значительно улучшить пока что доволь-

но скромную статистику решения данного задания: на ЕГЭ по информатике в 2021 и 2022 гг. с заданием справилось только около половины учеников – 48,2 и 49,3 % соответственно¹. Опрос, проведённый в одной из групп первокурсников факультета «Информатика, искусственный интеллект и системы управления» МГТУ им. Н.Э. Баумана, показал, что все студенты группы решали данное за-

дание с помощью программного кода, а несколько участников ЕГЭ по информатике ещё и проверили полученный на компьютере результат, проведя вычисления вручную. Кроме того, хотя автор также рекомендует проверять полученный ответ при помощи аналитического решения, приведённые методы требуют гораздо меньшего времени и усилий для решения задания.

¹ Попов В. С., Анализ изменения среднего процента выполнения заданий ЕГЭ по информатике в связи с введением компьютерного формата экзамена // Новые информационные технологии в образовании: Сборник научных трудов 13-й международной научно-практической конференции "Новые информационные технологии в образовании" 31 января-1 февраля 2023 г. / Под общ.ред. проф. Д.В. Чистова. Часть 2. – М.: ООО "IC-Паблишинг", 2023

НОВОСТИ

Ученые МФТИ научились отслеживать транспозоны растений в реальном времени



Ученые разработали новый метод для отслеживания мобильных генетических элементов у растений. Это участки ДНК, которые перемещаются по геному и вызывают изменение различных признаков (цвета, формы и других). В отличие от старых технологий, благодаря которым отследить такие перемещения удавалось только между разными поколениями (родительским и дочерним растением), новый подход позволяет выявлять их практически в режиме реального времени. Предложенный метод поможет ученым ускорить выведение новых линий сельскохозяйственных растений. Результаты исследования опубликованы в *Journal of Integrative Plant Biology*.

Предложенный метод поможет ученым ускорить выведение новых линий сельскохозяйственных растений. Результаты исследования опубликованы в *Journal of Integrative Plant Biology*.

Мобильные генетические элементы (или транспозоны) — это древний природный механизм, формирующий генетическое разнообразие всего живого на Земле. Они найдены почти во всех живых организмах и могут составлять до примерно 93% генома. Транспозоны представляют собой участки ДНК, способные перемещаться внутри одной клетки (или генома). Делая это они через молекулу РНК или через «вырезание» себя из генома. Во время таких перемещений транспозоны создают свои копии, тем самым изменяя последовательности цепочек ДНК, что влияет на многие признаки, например на цвет плодов и форму листьев современных сортов сельскохозяйственных культур. Поэтому мобильные элементы можно назвать одной из основных движущих сил разнообразия, адаптации и эволюции растений.

Если бы можно было контролировать их перемещение, появилась бы возможность создавать новые линии сельскохозяйственных растений. Однако даже просто отследить, куда переместятся транспозоны, ранее было возможно только сравнив геномы различных поколений — родительского растения и его потомков.