

Решение заданий №17 КЕГЭ
по информатике.

Проверка умений составлять
алгоритм на языке Python

Учитель информатики и ИКТ
лица «Армавирский классический
лицей»

Бабанская-Никель Ингрида Павловна

Из спецификации:

№	Проверяемые предметные требования к результатам освоения основной образовательной программы	Коды проверяемых элементов содержания (по кодификатору)	Коды проверяемых требований (по кодификатору)	Уровень сложности задания	Требуется использование специализированного программного обеспечения	Макс. балл за выполнение задания	Примерное время выполнения задания (мин.)
17	Умение составить алгоритм обработки числовой последовательности и записать его в виде простой программы (10–15 строк) на языке программирования	3.10	2.12	II	да	1	14

Раздел 2. Перечень элементов содержания, проверяемых на едином государственном экзамене по информатике

В таблице 3 приведён составленный на основе федеральной образовательной программы среднего общего образования по информатике перечень проверяемых элементов содержания.

Таблица 3

Код	Проверяемый элемент содержания	Уровень программы	Наличие данного элемента содержания в кодификаторе ЕГЭ
3.10	Массивы и последовательности чисел. Вычисление обобщённых характеристик элементов массива или числовой последовательности (суммы, произведения, среднего арифметического, минимального и максимального элементов, количества элементов, удовлетворяющих заданному условию). Линейный поиск заданного значения в массиве. Алгоритмы работы с элементами массива с однократным просмотром массива. Сортировка одномерного массива. Простые методы сортировки (метод пузырька, метод выбора, сортировка вставками). Сортировка слиянием. Быстрая сортировка массива (алгоритм QuickSort). Двоичный поиск в отсортированном массиве	БУ, УУ	+

В таблице 2 приведён составленный на основе п. 9.8 изменённого в 2022 г. ФГОС перечень проверяемых требований к предметным результатам освоения основной образовательной программы среднего общего образования. В таблице 2 показано, что требования к предметным результатам из изменённого в 2022 г. ФГОС являются преемственными и детализируют формулировки требований из ФГОС 2012 г.

Проверяемые требования к предметным результатам соотношены с метапредметными результатами (из таблицы 1).

Таблица 2

Код проверяемого требования	Проверяемые требования к предметным результатам освоения основной образовательной программы среднего общего образования на основе изменённого в 2022 г. ФГОС	Уровень предметных требований ФГОС	Метапредметный результат	Обобщённые формулировки требований к предметным результатам из ФГОС 2012 г.
2.12	предложена по улучшению программного кода Умение реализовывать на выбранном для изучения языке программирования высокого уровня (Паскаль, Python, Java, C++, C#) типовые алгоритмы обработки чисел, числовых последовательностей и массивов: представление числа в виде набора простых сомножителей; нахождение максимальной (минимальной) цифры натурального числа, записанного в системе счисления с основанием, не превышающим 10; вычисление обобщённых характеристик элементов массива или числовой	БУ, УУ	МП 1.2; 3.1; 3.2	Владение стандартными приёмами написания на алгоритмическом языке программы для решения стандартной задачи с использованием основных конструкций программирования и отладки таких программ (БУ); владение навыками и опытом разработки программ в выбранной среде программирования, включая тестирование и отладку программ (УУ)
	последовательности (суммы, произведения среднего арифметического, минимального и максимального элементов, количества элементов, удовлетворяющих заданному условию); сортировку элементов массива; умение использовать в программах данные различных типов с учётом ограничений на диапазон их возможных значений, применять при решении задач структуры данных (списки, словари, стеки, очереди, деревья); применять стандартные и собственные подпрограммы для обработки числовых данных и символьных строк; использовать при разработке программ библиотеки подпрограмм; умение использовать средства отладки программ в среде программирования			

Для решения задач надо знать и уметь работать:

- с циклами `for` и `while`,
- с несколькими условиями и применять операции `and`, `or`, `not`,
- находить остаток от деления (числа, которые делятся на 4 или на 7 и не делятся на 11, 17, 19 и 21),
- находить целую часть от деления (запись которых в шестеричной системе имеет не более 5 цифр и записывается на 13),

Для решения задач надо знать и уметь работать:

- с системами счисления, переводить из одной СС в другую (запись в двоичной и четверичной системах счисления заканчивается одинаковой цифрой или запись в шестнадцатеричной системе счисления заканчивается на 'FA'),
- находить максимальное или минимальное число в последовательности (алгоритм нахождения и функции в Python),
- модуль числа,
- уметь работать с файлами, со строками и списками.

Количество цифр в восьмеричной и десятичной записях числа не совпадает

Арифметическое вычисление	Используя строки
<pre>k10=0 p=i while p!=0: k10=k10+1 p=p//10 k8=0 p=i while p!=0: k8=k8+1 p=p//8 if k10!=k8:</pre>	<pre>if len(oct(i)[2:])!=len(str(i)):</pre>

Из демоверсии 2021 года

17

Рассматривается множество целых чисел, принадлежащих числовому отрезку $[1016; 7937]$, которые делятся на 3 и не делятся на 7, 17, 19, 27.

Найдите количество таких чисел и максимальное из них.

В ответе запишите два целых числа: сначала количество, затем максимальное число.

Для выполнения этого задания можно написать программу или воспользоваться редактором электронных таблиц.

Ответ:

--	--

Из демоверсии 2023года



Задание выполняется с использованием прилагаемых файлов.

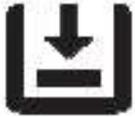
17

В файле содержится последовательность целых чисел. Элементы последовательности могут принимать целые значения от $-10\,000$ до $10\,000$ включительно. Определите количество пар последовательности, в которых только одно число оканчивается на 3, а сумма квадратов элементов пары не меньше квадрата максимального элемента последовательности, оканчивающегося на 3. В ответе запишите два числа: сначала количество найденных пар, затем максимальную из сумм квадратов элементов таких пар. В данной задаче под парой подразумевается два идущих подряд элемента последовательности.

Ответ:

--	--

Из проекта демоверсии 2024года



Задание выполняется с использованием прилагаемых файлов.

17

В файле содержится последовательность натуральных чисел, каждое из которых не превышает 100 000. Определите количество троек элементов последовательности, в которых ровно два из трёх элементов являются трёхзначными числами, а сумма элементов тройки не больше максимального элемента последовательности, оканчивающегося на 13. Гарантируется, что в последовательности есть хотя бы одно число, оканчивающееся на 13. В ответе запишите количество найденных троек чисел, затем максимальную из сумм элементов таких троек. В данной задаче под тройкой подразумевается три идущих подряд элемента последовательности.

Подключение файла

#ОБЫЧНОЕ СЧИТЫВАНИЕ

```
f=open('17.txt')           #открываем файл для чтения
a=[]                       #создаем список a
for i in f:                #проходим по каждой строке файла
    a.append(int(i))       #добавляем числа в список
print(a)                   #выводим список на экран
f.close()                  #закрываем файл
```

#ЧЕРЕЗ ГЕНЕРАТОР СПИСКОВ

```
with open('17.txt') as f:  #открываем файл для чтения
    a=[int(i) for i in f]  #используем генератор списков
print(a)                   #выводим список на экран
```

#ЧЕРЕЗ MAP

```
with open('17.txt') as f:  #открываем файл для чтения
    a=list(map(int, f))     #через map преобразуем строки в числа
print(a)                   #выводим список на экран
```

#дети делают в основном так

```
f=open("17-257.txt")
a=[int(x) for x in f.readlines()]
```

Перебираем рядом стоящие пары.
Значит промежуток индексов
списка будет $[0, \text{len}(a)-1]$

```
f=open("17-demo2023.txt")
a=list(map(int, f))
m3=-100000
for i in range(len(a)):
    if abs(a[i])%10==3: #считаем модуль числа так как числа отрицательные
        m3= max(m3, a[i])
print(m3) #Получается число 9973.

k=0
m=0
for i in range(len(a)-1):
    if ((abs(a[i])%10==3) and (abs(a[i+1])%10!=3) or \
        (abs(a[i])%10!=3) and (abs(a[i+1])%10==3)) \
        and (a[i]**2 + a[i+1]**2 >= m3**2):
        k=k+1
        m = max(m, a[i]**2 + a[i+1]**2 )
print(k, m)
```

Демо2023

Ответ: 180 190360573

Теперь тройки элементов.
Промежуток индексов списка
будет $[0, \text{len}(a)-2]$

```
f=open("17_2024.txt")
a=[int(x) for x in f.readlines()]
m3=-100000
for i in range(len(a)):
    if a[i]%100==13:
        m3= max(m3, a[i])
k=m=0
for i in range(len(a)-2):
    if a[i]+a[i+1]+a[i+2]<= m3:
        if (99<a[i]<1000)+(99<a[i+1]<1000)+(99<a[i+2]<1000)==2:
            k=k+1
            m = max(m, a[i]+a[i+1]+a[i+2])
print(k, m)
```

Проект демоверсии 2024 Ответ:959 97471

270) В файле 17-243.txt содержится последовательность целых чисел. Элементы последовательности могут принимать целые значения от 0 до 10 000 включительно. Определите количество пар чисел, в которых ровно один из двух элементов больше, чем сумма цифр всех чисел в файле, делящихся на 35, а шестнадцатеричная запись другого оканчивается на EF. В ответе запишите два числа: сначала количество найденных пар, а затем – минимальную сумму элементов таких пар. В данной задаче под парой подразумевается два идущих подряд элемента последовательности.

```
"""
```

```
f=open('17-243.txt')
a=[int(x) for x in f.readlines()]
k=0
s=999999999
m=0

for i in range(len(a)):
    if a[i]%35==0:
        p=a[i]
        while p>0:
            m=m+(p%10)
            p=p//10

for i in range(len(a)-1):
    a16=hex(a[i])
    a162=hex(a[i+1])
    if (a16[-1]=='f' and a16[-2]=='e' and (a[i]<=m) and (a[i+1]>m))\
        or (a162[-1]=='f' and a162[-2]=='e' and (a[i]>m) and (a[i+1]<=m)) :
        k=k+1
        s=min(s,a[i]+a[i+1])
print(k,s)
```

Ответ: 15 6410


```
"""
```

253) В файле 17-243.txt содержится последовательность целых чисел. Элементы последовательности могут принимать целые значения от 0 до 10 000 включительно. Определите количество пар чисел, в которых хотя бы один из двух элементов больше, чем наибольшее из всех чисел в файле, делящихся на 127, и в восьмеричной записи хотя бы одного элемента из двух содержится цепочка цифр 31. В ответе запишите два числа: сначала количество найденных пар, а затем – минимальную сумму элементов таких пар. В данной задаче под парой подразумевается два идущих подряд элемента последовате

```
"""
```

```
f=open('17-243.txt')
a=[int(x) for x in f.readlines()]
k=0
s=999999999
q=0
for i in range(len(a)):
    if a[i]%127==0:
        q=max(q, a[i])

for i in range(0, len(a)-1):
    l=0
    m=0
    if (a[i]>q or a[i+1]>q) and \
        (oct(a[i]).count("31")>0 or oct(a[i+1]).count("31")>0):
        k=k+1
        s=min(s, a[i]+a[i+1])
print(k, s)
```

Ответ: 28 11146

Среднего процента выполнения заданий



Рис 2. Средние проценты выполнения заданий

ЖОШНН							
17	Умение создавать собственные программы (20–40 строк) для обработки целочисленной информации	1.7.2	1.1.5	II	Да	1	20,5

Вывод об изменении успешности выполнения заданий (Краснодарский край)

Таблица 3-10

	B1	B2	B3	B4	B5	B6	B7	B8	B9	B10	B11
2023	93	83	78	84	36	19	63	29	26	82	61
2022	89	82	69	55	50	83	36	31	38	80	45
прирост	4,00	1,00	9,00	29,00	-14,00	-64,00	27,00	-2,00	-12,00	2,00	16,00

B12	B13	B14	B15	B16	B17	B18	B19	B20	B21	B22	B23
37	65	48	50	62	22	22	76	63	52	62	48
67	51	59	43	70	38	56	67	54	43	72	42
-30,00	14,00	-11,00	7,00	-8,00	-16,00	-34,00	9,00	9,00	9,00	-10,00	6,00

B24	B25	B26	B27
11	41	5	7
21	26	18	2
-10,00	15,00	-13,00	5,00

Если проанализировать три последних года выполнения задания 17, то можно увидеть, что сдали в Краснодарском крае

в 2021	– 59,25%,
в 2022	– 38%
в 2023	– 22%.

Что говорит о снижении результатов по данному заданию.

Рекомендации:

- необходимо уделить особое внимание практическому программированию, включая работу с файлами при вводе-выводе данных, работу с массивами, сортировку, обработку числовой и символьной информации;
- организации вычислений в электронных таблицах.

Рекомендации:

При выполнении заданий с развёрнутым ответом значительная часть ошибок экзаменуемых обусловлена недостаточным развитием у них таких метапредметных навыков, как **анализ условия задания, способность к самопроверке.**

Очевидно, что улучшение таких навыков будет способствовать существенно более высоким результатам ЕГЭ, в том числе и по информатике.

Рекомендации:

следует обратить особое внимание на развитие метапредметных способностей

- самостоятельно планировать способы достижения поставленных целей,
- находить эффективные пути достижения результата,
- уметь искать альтернативные нестандартные способы решения познавательных задач,
- а также к логическому мышлению.