

Подготовка к ОГЭ по информатике 15.2 задание, из опыта работы

Белаш М.А.¹, Мякшенко О.В.²

¹*mromanenko2@mail.ru*, ²*engschool30@yandex.ru*

МОУ «СОШ №30 с углубленным изучением отдельных предметов имени Героя Советского Союза Коваленко Петра Михайловича», Энгельс, Россия

Статья рассказывает об опыте работы подготовки к 15.2 заданию ОГЭ по информатике. Представлены минимальные требования к знаниям ученика для решения задания 15.2 на языке программирования Python. Предложен шаблон решения для типичных задач с сайта ФИПИ.

Ключевые слова: информатика, ОГЭ, Python.

15 задание ОГЭ по информатике – это задание на составление алгоритма, программы. Ученикам дается альтернатива. Можно составить алгоритм для Робота в среде Кумир – № 15.1, а можно написать программу на одном из языков программирования – № 15.2.

В школе преподают разные языки программирования. По программе это 8-9 класс. Когда стал вопрос, какому языку обучать, выбор пал на Python. По нашему мнению, этот язык имеет ряд преимуществ, он проще, удобнее. Он позволяет писать программы короче и понятнее. Выбирая язык для ОГЭ по информатике, нужно сразу подумать о том, что многие ученики, кто сдает информатику, пойдут в профиль, где углубленно будут изучать язык, который им понадобится для сдачи ЕГЭ. Поэтому считаем, что нужно 15 задание решать на Python.

Наша работа начинается с анализа заданий на сайте <https://fipi.ru/>. На сегодняшнее время были отобраны 38 задач. Порешав предложенные федеральным институтом педагогических измерений задания, пришли к выводу, что можно создать шаблон по решению заданий 15.

Минимальные требования к знаниям ученика

Ввод. Достаточно знать ввод целого числа.

Листинг 1. Оператор ввода

```
a = int (input())
```

Приложение будет ждать пока пользователь не введет с клавиатуры целочисленное значение переменной а.

1. Вывод.

Листинг 2. Оператор вывода

```
print (K)
```

Функция выведет на экран значение переменной К.

Если значение переменной К вещественное число с множеством знаков после запятой, например при выводе среднего значения, можно использовать форматирование.

Листинг 3. Оператор вывода с форматированием

```
print (" {:.1f} ".format(S/K))
```

Функция покажет значение деления S/K, с одним знаком после запятой.

2. Условный оператор.

Таблица 1

Неполное ветвление. Конструкция if	Полное ветвление. Конструкция if - else
if <условие>: инструкции 1 инструкция n	if <условие>: инструкции 1 инструкция n else: инструкции 1 инструкция n

Блок кода, который необходимо выполнить отделяется четырьмя пробелами слева.

3. Условия. Сложные условия.

Знаки сравнения: >, <, >=, <=, !=, ==

Листинг 4. Условия

```
a%8==0 {кратное 8}
a%2==0{четное}
a%10==8{оканчивается на 8}
a%10==0{оканчивается на 0}
```

При составлении сложных условий используем связки and – И, or – ИЛИ

4. Цикл while

Листинг 5. Цикл while

```
while <условие продолжения работы>:
    инструкции 1
    инструкция n
```

5. Цикл for

Листинг 6. Цикл for

```
for i in range (n):
    инструкции
```

где n – это число повторений цикла

Все задания сайта ФИПИ можно разделить на 2 вида.

1. Задания, которые решаются с помощью цикла while – на сайте ФИПИ их представлено 23 штуки. Это задания, где есть условие, что программа получает на вход целые числа, количество введённых чисел неизвестно, последовательность чисел заканчивается числом 0 (0 – признак окончания ввода, не входит в последовательность).

2. Задания, которые решаются с помощью цикла и for– на сайте ФИПИ их представлено 15 штук. Это задания, где есть условие, что программа получает на вход количество чисел в последовательности, а затем сами числа.

Цикл while

Нахождение количества.

Напишите программу, которая в последовательности целых чисел определяет количество чётных чисел, кратных 7. Программа получает на вход целые числа, количество введённых чисел неизвестно, последовательность чисел заканчивается числом 0 (0 – признак окончания ввода, не входит в последовательность). Количество чисел не превышает 1000. Введённые числа по модулю не превышают 30 000.

Таблица 2

Входные данные	Выходные данные
-32	1
14	
17	
0	

Листинг 7. Пример программы

```
a=int (input ())
k=0
while a!=0:
```

```

if a%2==0 and a%7==0 and a<=30000:
    K=K+1
a=int (input ())
print (K)

```

В остальных задачах меняется только условие, после слова if.

Нахождение суммы. Напишите программу, которая находит сумму в последовательности чисел. Числа в последовательности натуральные. Условие: кратные 7 (делится на 7 без остатка), оканчиваются на 0. Введенные числа меньше или равны 300.

Таблица 3

Входные данные	Выходные данные
14 140 20 70 0	210

Листинг 8. Пример программы

```

a=int (input ())
S=0
while a!=0:
    if a%7==0 and a%10==0 a<=300:
        S=S+a
    a=int (input ())
print (S)

```

В остальных задачах меняется только условие, после слова if.

Нахождение среднего арифметического

Напишите программу для последовательности чисел. Программа находит среднее арифметическое. Числа в последовательности натуральные. Условие: оканчиваются на 1. Если таких чисел нет, то пользователь получает сообщение 'NO'. Введенные числа меньше или равны 300.

Таблица 4

Входные данные	Выходные данные
21 30 11 0	16.0
16 5 0	NO

Листинг 9. Пример программы

```

a=int (input ())
S=0
K=0
while a!=0:
    if a%10==1 and a<=300:
        S=S+a
        K=K+1
    a=int (input ())

```

```

if K>0:
    print (" {:.1f}".format(S/K))
else:
    print ('NO')

```

В остальных задачах меняется только условие, после слова if.

Цикл for

Нахождение количества

Напишите программу, которая в последовательности натуральных чисел определяет количество чисел, кратных 3. Программа получает на вход количество чисел в последовательности, а затем сами числа. В последовательности всегда имеется число, кратное 3. Количество чисел не превышает 1000. Введённые числа не превышают 3000. Программа должна вывести одно число – количество чисел, кратных 3.

Таблица 5

Входные данные	Выходные данные
3	2
12	
26	
24	

Листинг 10. Пример программы

```

K=0
kol= int (input ())
for i in range(kol):
    a= int (input ())
    if a%3==0 and a<=300:
        K=K+1
print (K)

```

В остальных задачах меняется только условие, после слова if.

Нахождение суммы

Напишите программу для последовательности чисел. Программа находит сумму. Числа в последовательности натуральные. Условие: оканчиваются на 8. На вход программа получает в первую очередь количество чисел в последовательности, а затем сами числа, которые необходимо проверить. Введенные числа меньше или равны 300.

Таблица 6

Входные данные	Выходные данные
3	46
18	
28	
24	

Листинг 11. Пример программы

```

S=0
kol= int (input ())
for i in range(kol):
    a= int (input ())
    if a%10==8 and a<=300:
        S=S+a

```

```
print (S)
```

В остальных задачах меняется только условие, после слова if.

Нахождение максимального числа

Напишите программу для последовательности чисел. Программа находит максимальное значение. Числа в последовательности натуральные. Условие: оканчиваются на 9. На вход программа получает в первую очередь количество чисел в последовательности, а затем сами числа, которые необходимо проверить. Введенные числа меньше или равны 30000.

Таблица 7

Входные данные	Выходные данные
3	19
9	
19	
23	

Листинг 12. Пример программы

```
m=0
kol= int (input ())
for i in range(kol):
    a= int (input ())
    if a%10==9 and a>m and a<=30000:
        m=a
print (m)
```

В остальных задачах меняется только условие, после слова if.

Нахождение минимального числа

Напишите программу для последовательности чисел. Программа находит минимальное значение. Числа в последовательности натуральные. Условие: оканчиваются на 2. На вход программа получает в первую очередь количество чисел в последовательности, а затем сами числа, которые необходимо проверить. Введенные числа меньше или равны 30000.

Таблица 8

Входные данные	Выходные данные
3	12
22	
12	
36	

Листинг 13. Пример программы

```
n=3000
K=0
kol= int (input ())
for i in range(kol):
    a= int (input ())
    if a<n and a<=30000:
        n=a
print (n)
```

В остальных задачах меняется только условие, после слова if.

Нахождение среднего арифметического

Напишите программу для последовательности чисел. Программа находит среднее арифметическое чисел, которые подходят под условие задания. Числа в последовательности натуральные. Условие: числа больше 0. На вход программа получает в первую очередь количество чисел в последовательности, а затем сами числа, которые необходимо проверить. На выходе выводится среднееарифметическое и количество чисел, которые удовлетворяют условию.

Таблица 9

Входные данные	Выходные данные
4	10.0
-5	2
12	
-2	
8	

Листинг 14. Пример программы

```
K=0
S=0
kol= int (input ())
for i in range(kol):
    a= int (input ())
    if a>0:
        S=S+a
        K=K+1
print (" {:.1f} ".format (S/K))
print (K)
```

По нашему мнению, используя данные шаблоны, можно успешно подготовить учеников к сдаче экзамена по информатике.

Список литературы

[1] <https://fipi.ru/>