

ОГЭ-Информатика

5-8 задание

5 Задание

Краткое описание задания.

Данное задание посвящено тематике анализа алгоритмов для конкретного исполнителя с фиксированным набором команд.

Максимальный балл за данное задание - 1 балл;

Теория.

Исполнитель (например Сигма, Альфа и тп) - это программа, которая выполняет определенные команды для изменения данных на экране. Например, изменение числа - увеличение числа на 3, или там например разделить число на 5.

Популярные команды:

- **Прибавь 1** - увеличивает текущее число на экране на 1.
- **Раздели на b** - делит текущее число на b . (b - неизвестное натуральное число, $b \geq 2$).
- **Вычти 1** - уменьшает текущее число на экране на 1.
- **Умножь на 3** - умножает текущее число на 3.

Пример-1.

1 Тип 5 № 10390 

У исполнителя Сигма две команды, которым присвоены номера:

1. прибавь 1;
2. раздели на b

(b — неизвестное натуральное число; $b \geq 2$).

Выполняя первую из них, Сигма увеличивает число на экране на 1, а выполняя вторую, делит это число на b . Программа для исполнителя Сигма — это последовательность номеров команд. Известно, что программа 12111 переводит число 63 в число 11. Определите значение b .

12111

+1
:6
+1
+1
+1

63 → 11

$$\frac{63+1}{b} + 3 = 11$$

$$\frac{64}{b} = 8$$

$$b = 8$$

Пример-2.

19

Тип 5 № 16013 

У исполнителя Квадратор две команды, которым присвоены номера:

1. возведи в квадрат

2. прибавь 3

Первая из них возводит число на экране во вторую степень, вторая увеличивает его на 3.

Составьте алгоритм получения из числа 1 числа 25, содержащий не более 5 команд. В ответе запишите только номера команд.

(Например, 12221 — это алгоритм:

возведи в квадрат

прибавь 3

прибавь 3

прибавь 3

возведи в квадрат,

который преобразует число 2 в 169.)

Если таких алгоритмов более одного, то запишите любой из них.

$$1 \rightarrow 25$$

$$1 + 3 = 4 \quad (2)$$

$$4^2 = 16 \quad (1)$$

$$16 + 3 \cdot 3 = 16 + 9 = \boxed{25} \quad (2), (2), (2)$$

Ответ: 21222

Пример-3 (не 2024).

1 Тип Д6 № 6    

Исполнитель Чертежник перемещается на координатной плоскости, оставляя след в виде линии. Чертежник может выполнять команду **Сместиться на (a, b)** (где a, b — целые числа), перемещающую Чертежника из точки с координатами (x, y) в точку с координатами $(x + a, y + b)$. Если числа a, b положительные, значение соответствующей координаты увеличивается; если отрицательные, уменьшается.

Например, если Чертежник находится в точке с координатами $(4, 2)$, то команда Сместиться на $(2, -3)$ переместит Чертежника в точку $(6, -1)$.

Запись

Повтори k раз

Команда1 Команда2 Команда3

Конец

означает, что последовательность команд **Команда1 Команда2 Команда3** повторится **k** раз.

Чертежнику был дан для исполнения следующий алгоритм:

Повтори 2 раз

Команда1 Сместиться на (3, 2) Сместиться на (2, 1)

Конец

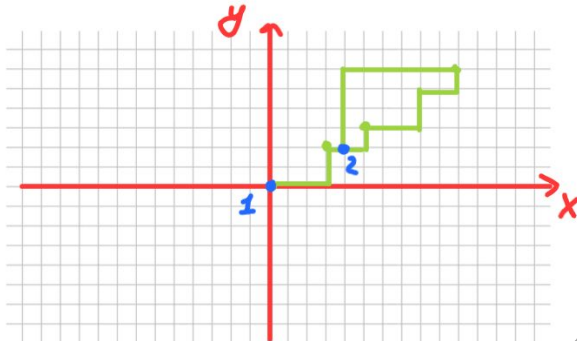
Сместиться на $(-6, -4)$

После выполнения этого алгоритма Чертежник вернулся в исходную точку. Какую команду надо поставить вместо команды **Команда1**?

- 1) Сместиться на $(-2, -1)$
- 2) Сместиться на $(1, 1)$
- 3) Сместиться на $(-4, -2)$
- 4) Сместиться на $(2, 1)$

$$y_{T.2} (4; 2) \rightarrow (-4; -2) = (-2; -1)$$

Ответ: 1



6 Задание

Краткое описание задания.

Данное задание посвящено тематике исполнения команд записанных на языке программирования. Как правило задание начинается с ключевой фразы “Ниже приведена программа, записанная на пяти языках программирования.”.

Максимальный балл за данное задание - 1 балл;

Теория.

Программирование - это процесс создания программ, которые дают компьютеру инструкции о том, что и как делать. Программы состоят из последовательности команд, написанных на языке программирования.

Язык программирования - это формальный язык, который состоит из набора правил для написания программ. С помощью этих языков программисты могут сообщать компьютеру, какие действия необходимо выполнить. Примеры языков программирования включают Python, Java, C++, JavaScript, и многие другие.

Переменная - это (хранилище данных, значений) контейнер для хранения данных, которые могут изменяться во время выполнения программы. Переменной присваивается имя, и с её помощью можно обращаться к данным в программе. Данные которые может хранить переменная - зависят от типа данных - это тип значения.

В Python существует несколько типов данных, включая:

- **int (целые числа):** например, 1, -3, 42.
- **float (вещественные числа):** например, 3.14, -0.001, 2.71828.
- **str (строки):** текст, например, "hello", "world".
- **bool (логические значения):** True или False.
- **None** - пустой тип;

Переменные. (с примером)

Переменные используются для хранения данных (значений). Каждая переменная имеет определенный тип данных (тип значения, например: целое число (*int*), вещественное число (*float*), строка (*str*), булевый (логический *True* или *False*) тип (*bool*) и др).

Пример создания переменных:

```
age = 18
```

```
height = 1.75
```

```
string = "Hello, world!"
```

```
is_student = True
```

- `age`, `height`, `string`, `is_student` - это переменные.
- `18` - целочисленный тип данных (*int*).
- `1.75` - тип данных с плавающей точкой (*float*).
- `True` - логический тип данных (*bool*).
- `"Hello, world!"` - тип данных строка (*str*).

Чтобы узнать тип данных переменной необходимо воспользоваться специальной функцией `type()`.
Например, `type(age)` или `type(string)`

Основные функции в Python.

- `input()` - функция для ввода данных с клавиатуры.
- `print()` - функция для вывода данных на экран.

Пример

```
name = input("Введите ваше имя: ")  
  
print("Привет, " + name + "!") # Привет, Алексей!  
  
print("Привет, ", name) # Привет, Алексей
```

- `name` - это переменная.

Примеры с print().

Простой вывод сообщение (текст или строка):

```
print("This is my test text!")
```

Простой вывод значения переменной (без доп текста):

```
a = 5
```

```
print(a)
```

Простой вывод значения переменной уже с текстом для красоты:

```
print("value is equal", a)
```

или так

```
print("value is equal " + str(a))
```

Пример с переносом строки (\n)

```
print("This is my test text!\n", "Hello world\n", "Read file\n")
```

Примеры с input().

Ввод целого числа с клавиатуры в переменную x.

```
x = int(input("Введите число: "))
```

Ввод вещественного числа (с плав.точкой).

```
float_x = float(input("Введите вещественное число: "))
```

Ввод строки (уже можно преобразование типа данных не делать!).

```
text = input("Введите текст: ")
```

Функция `INPUT` всегда **возвращает** строковый тип данных (str) именно поэтому мы и **не делаем преобразование** типа данных когда хотим ввести просто **строку** или **текст**.

Операции над переменными.

1) Арифметические операции:

- Сложение: +
- Вычитание: -
- Умножение: *
- Деление: /
- Целочисленное деление: //
- Остаток от деления: %
- Возведение в степень: **

Сравнительные операции:

Сравнительные операции (равно, неравно, больше, меньше и др):

```
x = 5
y = 10

# Операции сравнения
print(x == y) # False, равно
print(x != y) # True, не равно
print(x > y)  # False, больше
print(x < y)  # True, меньше
print(x >= y) # False, больше или равно
print(x <= y) # True, меньше или равно
```

True - истина (верно)

False - ложь (значит неверно)

Условный оператор (if-elif-else).

Условие - это как "если-то". В программировании или математике это способ проверить, выполняется ли какое-то условие, и в зависимости от этого делать разные вещи. Например, "если температура больше 25 градусов, то надень футболку, иначе возьми куртку".

То есть - **Условный оператор** в программировании представляет собой конструкцию, которая позволяет выполнять различные блоки кода в зависимости от выполнения определенного условия. Условие - выражение, которое вычисляется как True или False. Оно указывает, какое действие нужно выполнить.

Простой пример:

```
x = 5
if x > 10:
    print("x is greater than 10")
else:
    print("x is 10 or less")
```

Еще пример:

```
day = "Sunday"

if day == "Sunday":
    print("It's a holiday!")
elif day == "Saturday":
    print("It's the weekend!")
else:
    print("It's a weekday")
```


Объединение нескольких условий.

В программировании часто нужно проверять несколько условий одновременно. В Python для этого используются логические операторы: **and**, **or**, **not**.

Логические операторы

- **and** (и): Этот оператор возвращает **True**, если оба условия истинны.
- **or** (или): Этот оператор возвращает **True**, если хотя бы одно из условий истинно.
- **not** (не): Этот оператор возвращает **True**, если условие ложно, и **False**, если условие истинно.

```
if a > 0 and b > 0:
    print("Both numbers are positive.")

a = int(input())
b = int(input())
if a > 0 or b > 0:
    print("At least one number is positive.")

a = int(input())
if not a > 0:
    print("The number is not positive.")
```

Циклы.

- оператор **break** полностью выходит из цикла.
- оператор **continue** переходит к началу цикла (от слова продолжить)

Цикл в программировании - это инструкция, которая говорит компьютеру повторять выполнение определенного блока кода несколько раз. Мы используем циклы, чтобы сделать программы более эффективными и чтобы избежать повторения одних и тех же команд.

В Python есть два основных цикла:

- Цикл **while** выполняет блок кода (*набор команд*), пока условие ИСТИННО. Как только условие становится ЛОЖНЫМ, цикл завершается.
- Цикл **for** используется для перебора элементов в последовательности (*например, в списке*). Программа будет выполнять определенные действия для каждого элемента в последовательности.

Цикл while.

Работа цикла while



```
counter = 1
while counter <= 5:
    print(counter)
    counter += 1
```

Синтаксис цикла for.

```
for переменная in последовательность:  
    # Блок кода, который будет выполнен несколько раз
```

- **переменная** - это переменная, которая будет использоваться для доступа к элемент
- **последовательность** - это набор элементов, по которым будет выполняться цикл.

Цикл for.

Работа цикла for



`range()` - она генерирует список чисел, который обычно используется для работы с циклом `for`.

`range(2)` - значит список 0, 1

`range(5)` - значит список 0, 1, 2, 3, 4

`range(2, 5)` - значит уже от 2 начинаем т.е 2, 3, 4

`range(2, 5, 2)` - третий аргумент показывает насколько будет увеличена переменная `index` (стандартно это 1 - и единица не пишется, иными словами это `index += 1`)

Пример кода:

```
for index in range(3):  
    print("Привет, мир!")
```

В этом примере:

1. фраза Hello, world! выводится в консоль ровно три раза.
2. `range(3)` - значит пройти от позиции 0 до позиции 3 (не включая цифру 3) т.е три раза 0, 1, 2 кстати это значения `index` (счетчика) который переходит внутри цикла с каждой итерацией счетчик увеличился на 1

Пример-1.

Тип 6 № 10458

Ниже приведена программа, записанная на пяти языках программирования.

Бейсик	Python
<pre>DIM s, t AS INTEGER INPUT s INPUT t IF s > 10 OR t > 10 THEN PRINT 'YES' ELSE PRINT 'NO' ENDIF</pre>	<pre>s = int(input()) t = int(input()) if s > 10 or t > 10: print("YES") else: print("NO")</pre>
Паскаль	Алгоритмический язык
<pre>var s, t: integer; begin readln(s); readln(t); if (s > 10) or (t > 10) then writeln('YES') else writeln('NO') end.</pre>	<pre>алг нач цел s, t ввод s ввод t если s > 10 или t > 10 то вывод "YES" иначе вывод "NO" все кон</pre>
C++	
<pre>#include <iostream> using namespace std; int main() { int s, t; cin >> s; cin >> t; if (s > 10 t > 10) cout << "YES"; else cout << "NO"; return 0; }</pre>	

oge.sdamgia.ru

Было проведено 9 запусков программы, при которых в качестве значений переменных s и t вводились следующие пары чисел:

(1, 2); (11, 2); (1, 12); (11, 12); (-11, -12); (-11, 12); (-12, 11); (10, 10); (10, 5).

Сколько было запусков, при которых программа напечатала «YES»?

Ответ: 5

(1,2) -
(11,2) +
(1,12) +
(11,12) +
(-11,-12) -
(-11,12) +
(-12,11) +
(10,10) -
(10,5) -

Пример-2 (не 2024).

1 Тип Д7 № 8    

В программе «:=» обозначает оператор присваивания, знаки «+», «-», «*» и «/» — соответственно операции сложения, вычитания, умножения и деления. Правила выполнения операций и порядок действий соответствуют правилам арифметики. Определите значение переменной b после выполнения алгоритма:

```
a := 2
b := 4
a := 2*a + 3*b
b := a/2*b
```

В ответе укажите одно целое число — значение переменной b.

$$a = 2$$

$$b = 4$$

$$a = 2 \cdot 2 + 3 \cdot 4 = 4 + 12 = 16$$

$$b = \frac{16}{2} \cdot 4 = 8 \cdot 4 = 32$$

Ответ: 32

7 Задание

Краткое описание задания.

Данное задание на знание тем связанных с интернетом и файлами, а именно адресации в интернете. Задание охватывает такие термины как: url-адрес, ip-адрес, доменное имя, протоколы передачи данных http, https, ftp и др.

Максимальный балл за данное задание - 1 балл;

Теория.

URL (Uniform Resource Locator) - это уникальный адрес, который используется для доступа к ресурсам в интернете, таким как веб-страницы, изображения, файлы и т.д. URL состоит из нескольких частей, каждая из которых выполняет определенную функцию.

Основные компоненты URL

1. **Протокол:** Определяет метод доступа к ресурсу. Наиболее распространенные протоколы:
 - a. `http`: Hypertext Transfer Protocol, используется для передачи веб-страниц.
 - b. `https`: Secure Hypertext Transfer Protocol, защищенный вариант `http`.
 - c. `ftp`: File Transfer Protocol, используется для передачи файлов.
2. **Доменное имя:** Имя сервера, на котором находится ресурс. Примеры: `google.com`, `animal.ru`, `home.info`.
3. **Путь к ресурсу:** Указывает местоположение файла или страницы на сервере. Пример: `/path/to/file`. Пример: `/images/photo.jpg`.

Пример URL

Рассмотрим URL <http://example.com/path/to/page.html>:

- **`http`:** Протокол.
- **`example.com`:** Доменное имя.
- **`/path/to/page.html`:** Путь к ресурсу.

IP-адрес: Уникальный числовой адрес устройства в сети. Пример: `192.168.1.1`.

Пример-1.

Тип 7 № 698    

Доступ к файлу **fox.htm**, находящемуся на сервере **animal.ru**, осуществляется по протоколу **http**. Фрагменты адреса файла закодированы буквами от А до Ж. Запишите последовательность этих букв, кодирующую адрес указанного файла в сети Интернет.

- А) .htm
- Б) animal
- В) /
- Г) ://
- Д) http
- Е) .ru
- Ж) fox

Ж Г Б Е В Ж А
http://animal.ru/fox.htm

Ответ: ДГБЕВЖА

Пример-2.

5 Тип 7 № 498 

На месте преступления были обнаружены четыре обрывка бумаги. Следствие установило, что на них записаны фрагменты одного IP-адреса. Криминалисты обозначили эти фрагменты буквами А, Б, В и Г:

2.12	22	.30	5.121
А	Б	В	Г

Восстановите IP-адрес. В ответе укажите последовательность букв, обозначающих фрагменты, в порядке, соответствующем IP-адресу.

IP = 222.125.121.30

Ответ: БАГВ

Пример-3.

2

Тип 7 № [5276](#)    

Почтовый ящик **teacher** находится на сервере **shkola1.mos.ru**. В таблице фрагменты адреса электронной почты закодированы цифрами от 1 до 6. Запишите последовательность цифр, кодирующую этот адрес.

- 1) @
- 2) .ru
- 3) shkola1
- 4) .
- 5) teacher
- 6) mos

teacher@shkola1.mos.ru

Ответ: 513462

8 Задание

Краткое описание задания.

Данное задание связано с аспектами интернета - а именно поиск информации в интернете, выполнение поисковых запросов, множества и операции над множеством.

Максимальный балл за данное задание - 1 балл;

Теория.

Проще всего для решения этого задания использовать множества. **Множество** - это группа уникальных объектов. В множестве каждый элемент встречается только один раз.

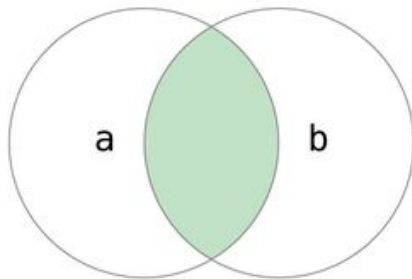
Примеры множеств

- Множество чисел меньше 5: {1, 2, 3, 4}
- Множество гласных букв в русском алфавите: {а, е, и, о, у}
- Множество цветов радуги: {красный, оранжевый, желтый, зеленый, голубой, синий, фиолетовый}

Операции над множеством.

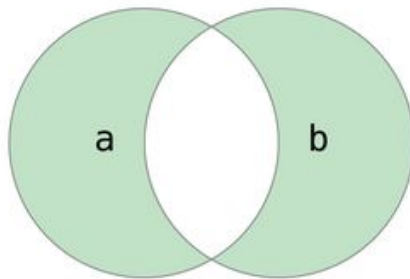
Пересечение

`a.intersection(b)`



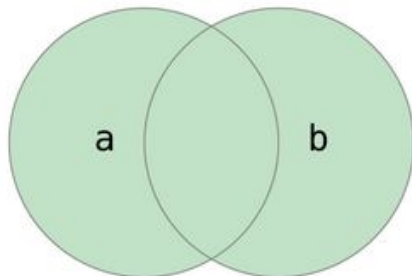
Симметрическая разность

`a.symmetricDifference(b)`



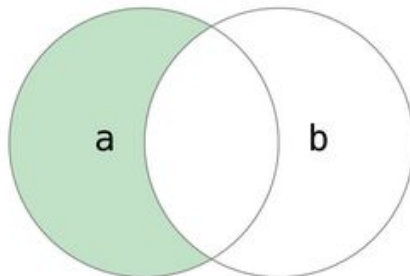
Объединение

`a.union(b)`



Разность

`a.subtract(b)`



- `&` - обозначается как “пересечение”;
- `|` - обозначается как “объединение”;

Пример-1.

15

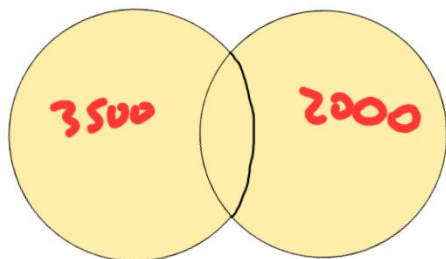
Тип 8 № 10486 

В языке запросов поискового сервера для обозначения логической операции «ИЛИ» используется символ «|», а для логической операции «И» — символ «&». В таблице приведены запросы и количество найденных по ним страниц некоторого сегмента сети Интернет.

Запрос	Найдено страниц (в тысячах)
Пушкин П	3500
Лермонтов Л	2000
Пушкин Лермонтов	4500 <small>ge.sdamgia.ru</small>

Какое количество страниц (в тысячах) будет найдено по запросу *Пушкин & Лермонтов*? Считается, что все запросы выполнялись практически одновременно, так что набор страниц, содержащих все искомые слова, не изменялся за время выполнения запросов.

П



Л

$$П + Л - (П | Л) = 1000$$

Пример-2.

41 Тип 8 № 10959 

В языке запросов поискового сервера для обозначения логической операции «ИЛИ» используется символ «|», а для логической операции «И» — символ «&».

В таблице приведены запросы и количество найденных по ним страниц некоторого сегмента сети Интернет.

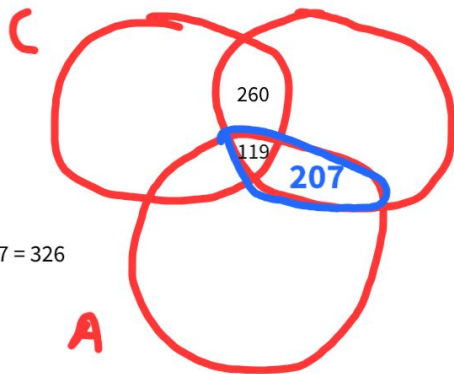
Запрос	Найдено страниц (в тысячах)
Сириус & Вега	260
Вега & (Сириус Арктур)	467
Сириус & Вега & Арктур	119

Какое количество страниц (в тысячах) будет найдено по запросу Вега & Арктур ? Считается, что все запросы выполнялись практически одновременно, так что набор страниц, содержащих все искомые слова, не изменялся за время выполнения запросов.

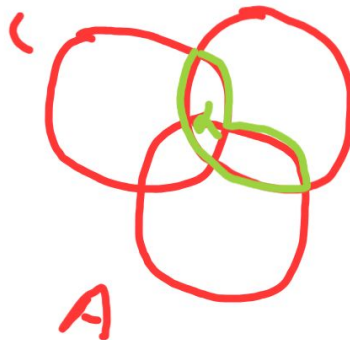
$$260 + x = 467$$
$$x = 207$$

значит:
Вега & Арктур = $119 + 207 = 326$

Ответ: 326



B



$$x = 467$$