Филиппов В.И., к.п.н.

ВОПРОСЫ МЕТОДИКИ ПОДГОТОВКИ ОБУЧАЮЩИХСЯ К ВЫПОЛНЕНИЮ ЗАДАНИЙ НА ОБРАБОТКУ СТРОК ПРИ ОРГАНИЗАЦИИ ОБУЧЕНИЯ ПРОГРАММИРОВАНИЮ В КУРСЕ ИНФОРМАТИКИ

Государственное бюджетное образовательное учреждение высшего образования Московской области «Академия социального управления», Московская область, г. Москва, vf95@rambler.ru

Filippov V.I., candidate of pedagogics

*QUESTIONS OF THE METHODOLOGY OF TRAINING STUDENTS TO PERFORM TASKS FOR PROCESSING STRINGS IN THE ORGANIZATION OF PROGRAMMING TRAINING IN THE COURSE OF COMPUTER SCIENCE*

Academy of public administration, Moscow region, Moscow, vf95@rambler.ru

Аннотация. В статье рассматриваются вопросы методики подготовки к выполнению заданий на обработку строк на языках программирования Pascal и Python при организации подготовки обучающихся к ГИА-11в форме ЕГЭ по информатике и ИКТ.

Abstract. The article deals with the issues of methods of preparation for performing tasks for processing strings in the programming languages Pascal and Python when organizing the preparation of students for the GIA-11 in the form of the Unified State Exam in computer science and ICT.

Ключевые слова: программирование, итоговая аттестация, информатика, обработка строк.

Key words: programming, final certification, computer science, string processing.

 В 2021 году Единый государственный экзамен по информатике и ИКТ впервые пройдет в компьютерной форме (К-ЕГЭ). Особенностями К-ЕГЭ по информатике и ИКТ являются:

1. Полностью автоматическая проверка заданий участников экзамена.
2. Наличие в контрольно-измерительных материалах практических заданий, которые участники экзамена выполняют с использованием прикладного программного обеспечения (редакторами электронных таблиц, текстовыми редакторами), а также средами программирования на языках: Школьный алгоритмический язык, С#, C++, Pascal, Java, Python.

Проведя анализ Спецификации контрольных измерительных материалов для проведения в 2021 году единого государственного экзамена по информатике и ИКТ и Кодификатора элементов содержания и требований к уровню подготовки выпускников образовательных организаций для проведения единого государственного экзамена по информатике и ИКТ следует отметить наличие заданий, в которых необходимо:

1. обрабатывать строки и символы;
2. проводить считывание информации для обработки (чисел и символов) из текстовых файлов;
3. применять в процессе обработки информации сортировку;
4. применять стандартные алгоритмы, определенные в Приложении №1 к Кодификатору элементов содержания и требований к уровню подготовки выпускников образовательных организаций для проведения единого государственного экзамена по информатике и ИКТ [2].

В 2022 году модель для проведения К-ЕГЭ по информатике и ИКТ будет усовершенствована. Могут появиться задания, ответом на которые будут являться программы на языках программирования. Данные задания планируется проверять через систему тестов.

Таким образом, возникает необходимость совершенствовать систему подготовки обучающихся по разделам «Основы алгоритмизации» и «Программирование» с учетом требований к подготовке выпускников, предъявляемых в ходе проведения ГИА-11 по информатике и ИКТ.

Далее в статье будет рассмотрена система заданий, направленная на подготовку обучающихся к выполнению задания на обработку строк и символов. Рекомендуется при проведении занятий использование языков программирования Pascal или Python. Конкретная среда программирования определяется учителем.

Согласно Спецификации контрольных измерительных материалов и Кодификатору элементов содержания и требований к уровню подготовки выпускников образовательных организаций для проведения в 2021 году единого государственного экзамена по информатике и ИКТ задание №24 КИМ К-ЕГЭ по информатике и ИКТ относится к высокому уровню сложности, планируемое время выполнения задания участником экзамена составляет 18 минут. Определены проверяемые элементы содержания: умение создавать собственные программы (10–20 строк) для обработки символьной информации и проверяемые умения: строить информационные модели объектов, систем и процессов в виде алгоритмов.

Общий вид задания можно представить следующим образом:

**Дано:**

В текстовом файле \*.txt находится цепочка из символов латинского алфавита ……………………….

**Найти:**

длину самой длинной подцепочки, состоящей из символов ……;

количество цепочек длины 3, удовлетворяющих следующим условиям;

длину самой длинной подцепочки, состоящей из одинаковых символов;

максимальное количество идущих подряд символов, среди которых каждые два соседних различны.

При создании обобщенного вида заданий использованы материалы Демонстрационного варианта для проведения К-ЕГЭ по информатике и ИКТ и прототипы задач, размещенные на сайте [https://kpolyakov.spb.ru/school/ege.htm [4](https://kpolyakov.spb.ru/school/ege.htm%20%5B4)**]. Т**екстовые файлы с заготовками заданий размещены на сайте проекта [4]. Возможны и другие варианты формулировок, так как задание относится к высокому уровню сложности и проверяет умения обучающихся применять полученные знания при решении заданий с новыми формулировками. Обращаем внимание, что результатом выполнения заданий является число, которое и вносится в электронный бланк участника экзамена.

Что необходимо знать участнику экзамена для решения данного задания?

***Первая подзадача*** – *чтение строки из файла.* В разных языках программирования это делается по-разному.

В языке **PascalABC.NET** связываем переменную input c файлом на компьютере \*.txt. Программа будет «думать», что читает данные, введённые с клавиатуры, а на самом деле эти данные будут прочитаны из файла \*.txt.

|  |
| --- |
| **assign( input, ‘\*.txt' );** |
| **readln(s);** |

В языке **FreePascal** необходимо дополнительно открывать входной поток с помощью процедуры reset(input):

**assign( input, ‘\*.txt' );**

**reset( input );**

**readln(s);**

В языке **Python** нужно открыть файл и прочитать все символы в символьную строку:

 **s = open(‘\*.txt').read()**

Рекомендуется в процессе решения после каждого завершенного этапа запускать программу и проверять ее работоспособность. Также важно убедиться при работе во Free Pascal, что в параметрах компилятора включена поддержка длинных символьных строк. На всякий случай стоит добавить в первой строке программы директиву {$H+} Перед тем, как запустить программу на выполнение, файл, с которым мы связываем переменную, должен быть обязательно загружен в ту папку, в которую мы сохраняем программу

***Вторая подзадача*** – *найти, длину цепочки символов в символьной строке s.*

Для успешного решения второй подзадачи участники экзамена должны знать:

1. Строка – это массив символов. К каждому символу можно обратиться в цикле. Длину цепочки символов можно определить с использованием функции *length (строка)* (Pascal) или *len (строка)* (Python).

Возможные реализации цикла с параметром:

**for** i:=1 **to** Length(s) (*Pascal)*

**for c in s:** или  **for i in range(len(s)): (***Python)*

1. Для проверки текущего символа на соответствие условию используем оператор ветвления. Возможные реализации проверки условий при поиске одного символа:

**if** (s[i]=’символ’) **then** (*Pascal)*

**if c==‘**символ**':** или **if c[i]==‘**символ**': (***Python)*

Возможные реализации проверки условий при поиске одного из нескольких символов:

 **if** (s[i] **in** ['A','C','E']) **then** (*Pascal)*

 **if с in** 'ACE': или **if s[i] in** 'ACE': **(***Python)*

1. Для проверки нескольких символов одновременно (например: текущего и следующего, предыдущего и текущего и так далее) вносим коррективы в цикл с параметром при просмотре, а также обращаемся к элементам строки с учетом ситуации.

Первый вариант. Два символа: текущий и следующий. В параметрах цикла вносим length (s)-1 (*Pascal)* или len(s)-1 **(***Python).* При построении условий используем индексы элементов s[i] и s[i-1].

Второй вариант. Два символа: текущий и предыдущий. В параметрах цикла устанавливаем просмотр со второго элемента: **for** i:=2 **to** Length(s) (*Pascal)* или с первого элемента: **for i in range(1,len(s)): (***Python)*

Для успешного решения задания участник экзамена должен разработать и самостоятельно реализовать фрагмент программы, осуществляющий подсчет требуемых элементов.

Далее в статье приводится разбор двух задач на обработку строк.

Задача №1. (Демонстрационный вариант К-ЕГЭ по информатике и ИКТ, 2021 г.) Текстовый файл состоит не более чем из 106 символов X, Y и Z. Определите максимальное количество идущих подряд символов, среди которых каждые два соседних различны. Для выполнения этого задания следует написать программу.

Программа на Python.

s = open('z24.txt').read()

count=1

maxCount=1

for i in range(len(s)-1):

 if s[i]!=s[i+1]:

 count = count+1

 else:

 if count>maxCount:

 maxCount=count

 count = 1

print(maxCount)

Программа на Pascal.

 var s: string;

 var count, maxCount,i: integer;

 begin

 assign(input,'z24.txt');

 readln(s);

 count:=1;

 maxCount:=1;

 for i:=1 to Length(s)-1 do

 if (s[i]<>s[i+1]) then

 count := count+1

 else

 begin

 if count>maxCount then maxCount:=count;

 count :=1;

 end;

 writeln(maxCount);

 end.

Задача 2. Источник: [4]. В текстовом файле k7c-2.txt находится цепочка из символов латинского алфавита A, B, C, D, E, F. Найдите количество цепочек длины 3, удовлетворяющих следующим условиям:

* 1-й символ – один из A, C, E;
* 2-й символ – один из A, D, F, который не совпадает с первым;
* 3-й символ – один из A, B, F, который не совпадает со вторым.

Программа на Python.

s = open('k7c-2.txt').read()

count = 0

for i in range(len(s)-2):

 if s[i] in 'ACE' and s[i+1] in 'ADF' \

 and s[i+2] in 'ABF' and s[i]!=s[i+1] \

 and s[i+1]!=s[i+2]:

 count += 1

print(count)

Программа на Pascal.

begin

 var s: string;

 var count: integer;

 assign(input,'k7c-2.txt');

 readln(s);

 count:=0;

 for var i:=1 to length(s)-2 do

 if (s[i] in ['A','C','E']) and

 (s[i+1] in ['A','D','F']) and

 (s[i+2] in ['A','B', 'F']) and

 (s[i]<>s[i+1])and

 (s[i+1]<>s[i+2]) then

 count := count+1;

 writeln(count);

end.

Итак, можно определить обобщенный алгоритм программы на языке программирования, реализующий задачу на обработку строк.

***В PascalABCNET***

* Описываем переменные.
* Переключаем входной поток с консоли на нужный файл, читаем одну строку из файла в строковую переменную :

**assign( input, ‘\*.txt' );**

**readln(s);**

* В цикле перебираем все символы строки **s** и выполняем обработку символа или символов:

**for i:=1 to Length(s) do**

 **обработать s[i]**

* Выводим результат:

**writeln (n)**

***В Python***

* Открываем файл и считываем все символы в символьную строку:

 **s = open(‘\*.txt').read()**

* В цикле перебираем все символы строки **s**  и выполняем обработку символа или символов.

**for i in range (len(s)):**

**обработать s[i]**

* Выводим результат^

**print (n)**

Планируемым результатом применения описанной выше системы заданий будет повышение результативности выполнения заданий на обработку строк, представленных в КИМ К-ЕГЭ по информатике и ИКТ.

Литература

1. Спецификация контрольных измерительных материалов для проведения в 2021 году единого государственного экзамена по информатике и ИКТ
2. Кодификатор элементов содержания и требований к уровню подготовки выпускников образовательных организаций для проведения единого государственного экзамена по информатике и ИКТ
3. Златопольский Д.М. 1400 задач по программированию // М: ДМК-Пресс 2020. – 192с.
4. К.Ю. Поляков Материалы для подготовки к ЕГЭ, https://kpolyakov.spb.ru/school/ege.htm