

16. Исполнение алгоритма, обрабатывающего цепочки символов или списки.

1. Некоторый алгоритм из одной цепочки символов получает новую цепочку следующим образом. Сначала вычисляется длина исходной цепочки символов; если она нечётна, то после последнего символа цепочки добавляется средний символ, а если чётна, то в середину цепочки символов вставляется символ **1**. В полученной цепочке символов каждая цифра заменяется на предыдущую (**1** заменяется на **0**, **2** – на **1**, и т. д., а **0** заменяется на **9**).

Получившаяся таким образом цепочка является результатом работы алгоритма.

Например, если исходной цепочкой была цепочка **456**, то результатом работы алгоритма будет цепочка **3454**, а если исходной цепочкой была **5203**, то результатом работы алгоритма будет цепочка **41092**.

Дана цепочка символов **4732**. Какая цепочка символов получится, если к данной цепочке применить описанный алгоритм дважды (то есть применить алгоритм к данной цепочке, а затем к результату вновь применить алгоритм)?

2. Некоторый алгоритм из одной цепочки символов получает новую цепочку следующим образом. Сначала вычисляется длина исходной цепочки символов; если она нечётна, то из исходной цепочки символов удаляется средний символ, а если чётна, то из строки удаляется первый символ. В полученной строке каждая цифра заменяется на следующую (**1** заменяется на **2**, **2** – на **3**, и т. д., а **9** заменяется на **0**).

Получившаяся таким образом цепочка является результатом работы алгоритма.

Например, если исходной цепочкой была цепочка **35682**, то результатом работы алгоритма будет цепочка **4693**, а если исходной цепочкой была **5193**, то результатом работы алгоритма будет цепочка **204**.

Дана цепочка символов **132685**. Какая цепочка символов получится, если к данной цепочке применить описанный алгоритм дважды (то есть применить алгоритм к данной цепочке, а затем к результату вновь применить алгоритм)?

3. Некоторый алгоритм из одной цепочки символов получает новую цепочку следующим образом. Сначала вычисляется длина исходной цепочки символов; если она нечётна, то в исходной цепочке символов удваивается средний символ, а если чётна, то в начало цепочки добавляется символ **1**.

В полученной строке каждая цифра заменяется на следующую (**1** заменяется на **2**, **2** – на **3**, и т. д., а **9** заменяется на **0**).

Получившаяся таким образом цепочка является результатом работы алгоритма.

Например, если исходной цепочкой была цепочка **356**, то результатом работы алгоритма будет цепочка **4667**, а если исходной цепочкой была **52**, то результатом работы алгоритма будет цепочка **263**.

Дана цепочка символов **35842**. Какая цепочка символов получится, если к данной цепочке применить описанный алгоритм дважды (то есть применить алгоритм к данной цепочке, а затем к результату вновь применить алгоритм)?

4. Некоторый алгоритм из одной цепочки символов получает новую цепочку следующим образом. Сначала вычисляется длина исходной цепочки символов; если она нечётна, то после последнего символа цепочки добавляется первый символ, а если чётна, то в начало цепочки добавляется последний символ. В полученной цепочке символов каждая цифра заменяется на предыдущую (**1** заменяется на **0**, **2** – на **1**, и т. д., а **0** заменяется на **9**). Получившаяся таким образом цепочка является результатом работы алгоритма.

Например, если исходной цепочкой была цепочка **483**, то результатом работы

алгоритма будет цепочка **3723**, а если исходной цепочкой была **3465**, то результатом работы алгоритма будет цепочка **42354**.

Дана цепочка символов **2974**. Какая цепочка символов получится, если к данной цепочке применить описанный алгоритм дважды (то есть применить алгоритм к данной цепочке, а затем к результату вновь применить алгоритм)?

5. Некоторый алгоритм из одной цепочки символов получает новую цепочку следующим образом. Если цепочка символов заканчивается буквой, то в конец цепочки добавляется **5**. Если цепочка символов заканчивается цифрой, то из цепочки удаляется последний символ. В полученной цепочке символов каждая буква, стоящая перед цифрой, заменяется буквой, следующей за ней в русском алфавите (**А** – на **Б**, **Б** – на **В**, и т. д., а **Я** – на **А**).

Получившаяся таким образом цепочка является результатом работы алгоритма.

Например, если исходной цепочкой была цепочка **Б2Р**, то результатом работы алгоритма будет цепочка **В2С5**, а если исходной цепочкой была **А1Д4**, то результатом работы алгоритма будет цепочка **Б1Д**.

Дана цепочка символов **П17Ф9В1**. Какая цепочка символов получится, если к данной цепочке применить описанный алгоритм дважды (то есть применить алгоритм к данной цепочке, а затем к результату вновь применить алгоритм)?

Русский алфавит: **АБВГДЕЁЖЗИЙКЛМНОПРСТУФХЦЧШЩЪЫЬЭЮЯ**

6. Некоторый алгоритм из одной цепочки символов получает новую цепочку следующим образом. Если количество цифр в цепочке символов больше количества букв, то в начало цепочки добавляется **А**. В противном случае в конец цепочки добавляется **1**. В полученной цепочке символов каждая буква, стоящая после цифры, заменяется буквой, следующей за ней в русском алфавите (**А** – на **Б**, **Б** – на **В**, и т. д., а **Я** – на **А**).

Получившаяся таким образом цепочка является результатом работы алгоритма.

Например, если исходной цепочкой была цепочка **4В2**, то результатом работы алгоритма будет цепочка **А4Г2**, а если исходной цепочкой была **М5Р3**, то результатом работы алгоритма будет цепочка **М5С31**.

Дана цепочка символов **2К8Т4**. Какая цепочка символов получится, если к данной цепочке применить описанный алгоритм дважды (то есть применить алгоритм к данной цепочке, а затем к результату вновь применить алгоритм)?

Русский алфавит: **АБВГДЕЁЖЗИЙКЛМНОПРСТУФХЦЧШЩЪЫЬЭЮЯ**

7. Некоторый алгоритм из одной цепочки символов получает новую цепочку следующим образом. Если цепочка символов начинается с буквы, то в начало и в конец цепочки добавляется **9**. В противном случае из цепочки удаляется первый символ. В полученной цепочке символов каждая буква заменяется буквой, стоящей перед ней в русском алфавите (**Б** – на **А**, **В** – на **Б**, и т. д., а **А** – на **Я**).

Получившаяся таким образом цепочка является результатом работы алгоритма.

Например, если исходной цепочкой была цепочка **Т5**, то результатом работы алгоритма будет цепочка **9С59**, а если исходной цепочкой была **5Г3**, то результатом работы алгоритма будет цепочка **В3**.

Дана цепочка символов **Д1М6У**. Какая цепочка символов получится, если к данной цепочке применить описанный алгоритм дважды (то есть применить алгоритм к данной цепочке, а затем к результату вновь применить алгоритм)?

Русский алфавит: **АБВГДЕЁЖЗИЙКЛМНОПРСТУФХЦЧШЩЪЫЬЭЮЯ**

8. Некоторый алгоритм из одной цепочки символов получает новую цепочку следующим образом. Если количество цифр в цепочке чётно, то в начало цепочки добавляется **1**. В противном случае из цепочки удаляется последняя цифра. В полученной цепочке символов каждая буква заменяется буквой, стоящей перед ней в русском алфавите (**Б** – на **А**, **В** – на **Б** и т. д., а **А** – на **Я**). Получившаяся таким образом цепочка является результатом работы алгоритма. Например, если исходной цепочкой была цепочка **Д75**, то результатом работы алгоритма будет цепочка **1Г75**, а если исходной цепочкой была **513Г**, то результатом работы алгоритма будет цепочка **51В**.
Дана цепочка символов **С2Д5М**. Какая цепочка символов получится, если к данной цепочке применить описанный алгоритм дважды (то есть применить алгоритм к данной цепочке, а затем к результату вновь применить алгоритм)?
Русский алфавит: **АБВГДЕЁЖЗИЙКЛМНОПРСТУФХЦЧШЩЪЫЬЭЮЯ**
9. Некоторый алгоритм из одной цепочки символов получает новую цепочку следующим образом. Сначала вычисляется длина исходной цепочки символов; если она чётна, то в начало цепочки добавляется символ **Z**, а если нечётна – дублируется символ в середине цепочки.
В полученной цепочке символов каждая буква заменяется буквой, стоящей перед ней в латинском алфавите (**В** заменяется на **А**, **С** заменяется на **В** и т. д., **А** заменяется на **Z**). Получившаяся таким образом цепочка является результатом работы алгоритма. Например, если исходной цепочкой была цепочка **ABCD**, то результатом работы алгоритма будет цепочка **YZABC**, а если исходной цепочкой была **MNO**, то результатом работы алгоритма будет цепочка **LMMN**.
Дана цепочка символов **PASCAL**. Какая цепочка символов получится, если к данной цепочке применить описанный алгоритм дважды (то есть применить алгоритм к данной цепочке, а затем к результату вновь применить алгоритм)?
Латинский алфавит: **ABCDEFGHIJKLMNOPQRSTUVWXYZ**
10. Некоторый алгоритм из одной цепочки символов получает новую цепочку следующим образом. Сначала вычисляется длина исходной цепочки символов; если она чётна, то в конец цепочки добавляется символ **A**, а если нечётна – удаляется средний символ цепочки.
В полученной цепочке символов каждая буква заменяется буквой, стоящей после нее в латинском алфавите (**А** заменяется на **В**, **В** заменяется на **С** и т. д., **Z** заменяется на **А**). Получившаяся таким образом цепочка является результатом работы алгоритма. Например, если исходной цепочкой была цепочка **ABCD**, то результатом работы алгоритма будет цепочка **BCDEB**, а если исходной цепочкой была **DFGHJ**, то результатом работы алгоритма будет цепочка **EGIK**.
Дана цепочка символов **BASIC**. Какая цепочка символов получится, если к данной цепочке применить описанный алгоритм дважды (то есть применить алгоритм к данной цепочке, а затем к результату вновь применить алгоритм)?
Латинский алфавит: **ABCDEFGHIJKLMNOPQRSTUVWXYZ**