

№1123. Зарплата

Орлов М.

June 2022

1 Описание задачи

Все сотрудники СКБ Контур любят получать зарплату. Часто и помногу. Однако руководство компании придерживается несколько иного мнения и выдаёт деньги строго раз в месяц. Посоветавшись, сотрудники решили, что если один из параметров (частота выдачи денег) зафиксирован, то можно попробовать изменить второй (количество выдаваемых денег) и придумали комбинацию в духе небезызвестного потомка янычар О.Бендера. Группа сотрудников, гордо называющих себя выпускниками математикомеханического факультета УрГУ, идёт к начальству и, ссылаясь на свой математический авторитет, доказывает, что компьютеры в бухгалтерии СКБ Контур будут работать гораздо эффективней, если зарплаты всех сотрудников примут вид палиндромов. Напомним, что числовой палиндром — это число, которое не изменяется при прочтении его справа налево. Например, 12344544321 — это палиндром, а 12345543210 — нет. Разумеется, начальство было вынуждено согласиться, но с одной оговоркой. Каждый сотрудник должен был сам пересчитать свою зарплату так, чтобы она приняла вид палиндрома, большего или равного исходной зарплате и, естественно, минимально возможного. Перед вами стоит нелёгкая задача — помочь сотрудникам компании СКБ Контур в их благородной борьбе за денежные знаки.

Исходные данные состоят из одной строки — исходной зарплаты сотрудника. Длина строки не превышает 2001 символа.

Результат должен состоять также из одной строки — новой зарплаты, вычисленной по вышеописанным правилам.

2 Цели задачи

Нам дано число на входном потоке. Мы должны вывести число на экран, которое является наименьшим палиндромом из тех, что больше заданного числа.

3 Методы решения задачи

Для решения задачи будем использовать Python 3.8, так как это достаточно простой язык в использовании, а наша задача не требует более специализированных эффективных языков. На входном потоке мы считываем строку символов(все символы - десятичные числа). Присваиваем ее длину некоторой переменной. Далее идет следующий алгоритм: Идем с начала строки до ее середины проверяя условия: 1. Если i -тое число с начала больше i -ого числа с конца, второе заменяем первым 2. Если, наоборот, меньше - проверяем символьную часть перед i -ым числом с конца на наличие девяток(высшие разряды после i -ого с конца разряда), заменяем девятки нулями, а разряд после двоек повышаем на единицу. После этого, как в первом пункте, заменяем i -ое число с конца i -ым числом с начала Выводим число на экран.

4 Код задачи

```
s=str(input())
n=len(s)
for i in range(0,int(n/2)):
    if s[i]>s[n-1-i]:
        s=s[:n-1-i]+s[i]+s[n-i:]
    elif s[i]<s[n-1-i]:
        k=1
        while s[n-1-i-k]=='9':
            s=s[:n-1-k-i]+'0'+s[n-k-i:]
            k=k+1
        s=s[:n-1-k-i]+str(int(s[n-1-k-i])+1)+s[n-k-i:]
        s=s[:n-1-i]+s[i]+s[n-i:]
print(s)
```

5 Результаты решения

С первого раза мне удалось сдать задачу. Она прошла все тесты. Время выполнения работы - 0.078. Выделено памяти - 676кб

6 Ссылка на git-репозиторий

<https://github.com/OrlovMlc/1123>