2021

177) Текстовый файл 24-157.txt содержит только заглавные буквы латинского алфавита (ABC…Z). Определите максимальное количество идущих подряд символов, среди которых нет стоящих рядом букв P и R (в любом порядке).

Вариант 1

**with** open(**'24data/24-157.txt'**) **as** f:  
 s = f.read().strip()  
  
pos = -1  
max\_len = 0  
**for** i **in** range(len(s) - 1):  
 **if** s[i] + s[i + 1] **in** (**'PR'**, **'RP'**):  
 max\_len = max(max\_len, i - pos)  
 pos = i  
max\_len = max(max\_len, len(s) - 1 - pos)  
print(max\_len)

Вариант 2

**with** open(**'24data/24-157.txt'**) **as** f:  
 s = f.read().strip()  
  
s = s.replace(**'PR'**, **'P R'**).replace(**'RP'**, **'R P'**)  
max\_seq = max(s.split(), key=len)  
print(len(max\_seq))

Ответ: 2940

2022

Досрок

Текстовый файл состоит из символов A, B и C. Определите максимальное количество идущих подряд пар символов AB или CB в прилагаемом файле. Искомая подпоследовательность должна состоять только из пар AB, или только из пар CB, или только из пар AB и CB в произвольном порядке следования этих пар.

Вариант 1

**with** open(**'24.txt'**) **as** f:  
 s = f.read().strip()  
  
max\_count = 0  
count = 0  
i = 0  
**while** i < len(s) - 1:  
 **if** s[i] **in** **'AC'** **and** s[i + 1] == **'B'**:  
 count += 1  
 i += 2  
 **else**:  
 count = 0  
 i += 1  
 max\_count = max(count, max\_count)  
print(max\_count)

Вариант 2

**with** open(**'24.txt'**) **as** f:  
 s = f.read().strip().replace(**'AB'**, **'1'**).replace(**'CB'**, **'1'**)  
count = 1  
**while '1'** \* count **in** s:  
 count += 1  
print(count - 1)

Вариант 3

**import** re  
  
**with** open(**'24.txt'**) **as** f:  
 s = f.read().strip()  
  
pattern = **r'(?:[AC]B)+'**max\_seq = max(re.findall(pattern, s), key=len)  
print(len(max\_seq) // 2)

Ответ: 65

2022 NPO PNO

213) Текстовый файл 24-213.txt содержит строку из символов N, O и P, всего не более чем из 106 символов. Определите максимальное количество идущих подряд последовательностей символов NРО или РNО в прилагаемом файле. Искомая подпоследовательность должна состоять только из троек NРО или только из троек РNО или только из троек NРО и РNО в произвольном порядке их следования.

Вариант 1

**with** open(**'24data/24-213.txt'**) **as** f:  
 s = f.read().strip()  
  
max\_len = 0  
cur\_len = 0  
**for** start **in** range(3):  
 **for** i **in** range(start, len(s), 3):  
 **if** s[i:i+3] **in** (**'NPO'**, **'PNO'**):  
 cur\_len += 1  
 max\_len = max(max\_len, cur\_len)  
 **else**:  
 cur\_len = 0  
print(max\_len)

Вариант 2

**with** open(**'24data/24-213.txt'**) **as** f:  
 s = f.read().strip()  
  
s = s.replace(**'NPO'**, **'1'**).replace(**'PNO'**, **'1'**)  
**for** letter **in 'NOP'**:  
 s = s.replace(letter, **' '**)  
max\_len = len(max(s.split()))  
print(max\_len)

Вариант 3

**import** re  
  
**with** open(**'24data/24-213.txt'**) **as** f:  
 s = f.read().strip()  
  
pattern = **r'(?:PNO|NPO)+'**max\_seq = max(re.findall(pattern, s), key=len)  
print(len(max\_seq) // 3)

Ответ: 297

2022 Согласная + гласная

212) Текстовый файл 24-212.txt содержит строку из символов A, B, C, D и O, всего не более чем из 106 символов. Определите максимальное количество идущих подряд пар символов вида «согласная + гласная».

Вариант 1

**with** open(**'24data/24-212.txt'**) **as** f:  
 s = f.read().strip()  
  
max\_len = 0  
cur\_len = 0  
**for** start **in** (0, 1):  
 **for** i **in** range(start, len(s) - 1, 2):  
 **if** s[i] **in 'BCD' and** s[i + 1] **in 'AO'**:  
 cur\_len += 1  
 max\_len = max(max\_len, cur\_len)  
 **else**:  
 cur\_len = 0  
print(max\_len)

Вариант 2

**import** re  
  
**with** open(**'24data/24-212.txt'**) **as** f:  
 s = f.read().strip()  
  
pattern = **r'(?:[BCD][AO])+'**max\_seq = max(re.findall(pattern, s), key=len)  
print(len(max\_seq) // 2)

Ответ: 202

2023 Досрок

Текстовый файл состоит не более, чем из 1 200 000 символов английского алфавита. Определите максимальное количество идущих подряд символов, среди которых символы Q, R, S в различных комбинациях (с учётом повторений) не стоят рядом.

Вариант 1

**from** itertools **import** product  
  
**with** open(**'24\_Досрок23.txt'**) **as** f:  
 s = f.read()  
tabu = list(map(**lambda** pair: **''**.join(pair), product(**'QRS'**, repeat=2)))  
  
max\_count = 0  
count = 1  
**for** i **in** range(1, len(s)):  
 **if** s[i - 1] + s[i] **in** tabu:  
 count = 1  
 **else**:  
 count += 1  
 max\_count = max(count, max\_count)  
print(max\_count)

Вариант 2

**import** re  
  
**with** open(**'2023\_досрок.txt'**) **as** f:  
 s = f.read().strip()  
  
s = re.sub(**r'([QRS])([QRS])'**, **r'\1 \2'**, s)  
s = re.sub(**r'([QRS])([QRS])'**, **r'\1 \2'**, s)  
max\_seq = max(s.split(), key=len)  
print(len(max\_seq))

Ответ: 544

2023 Y150

265) Текстовый файл 24-263.txt состоит не более чем из 106 символов и содержит только заглавные буквы латинского алфавита. Определите максимальную длину подстроки, в которой символ Y встречается не более 150 раз.

Вариант 1

**with** open(**'24data/24-263.txt'**) **as** f:  
 s = f.read().strip()  
positions = [-1]  
**for** i **in** range(len(s)):  
 **if** s[i] == **'Y'**:  
 positions.append(i)  
positions.append(len(s))  
max\_len = 0  
**for** i **in** range(151, len(positions)):  
 cur\_len = positions[i] - positions[i - 151] - 1  
 max\_len = max(cur\_len, max\_len)  
print(max\_len)

Вариант 2

**with** open(**'24-263.txt'**) **as** f:  
 s = f.read().strip()  
lines = s.split(**'Y'**)  
max\_len = 0  
**for** i **in** range(len(lines) - 150):  
 cur\_len = len(**'Y'**.join(lines[i:i+151]))  
 max\_len = max(max\_len, cur\_len)  
print(max\_len)

Ответ: 5195

2023 16СС

267) Текстовый файл 24-264.txt состоит не более чем из 106 символов и содержит только заглавные буквы латинского алфавита и цифры. Определите максимальную длину подстроки, которая может являться записью числа в шестнадцатеричной системе счисления.

Вариант 1

**from** string **import** ascii\_uppercase  
  
**with** open(**'24data/24-264.txt'**) **as** f:  
 s = f.read().strip()

not\_hex\_digits = ascii\_uppercase[6:]  
max\_len = 0  
cur\_len = 0  
is\_number = **False  
for** c **in** s:  
 **if** is\_number:  
 **if** c **in** not\_hex\_digits:  
 cur\_len = 0  
 is\_number = **False  
 else**:  
 cur\_len += 1  
 **else**:  
 **if** c **in** not\_hex\_digits **or** c == **'0'**:  
 cur\_len = 0  
 **else**:  
 cur\_len = 1  
 is\_number = **True** max\_len = max(max\_len, cur\_len)  
print(max\_len)

Вариант 2

**from** string **import** ascii\_uppercase  
  
**with** open(**'24data/24-264.txt'**) **as** f:  
 s = f.read().strip()  
s1 = s  
not\_hex\_digits = ascii\_uppercase[6:]  
**for** c **in** not\_hex\_digits:  
 s1 = s1.replace(c, **' '**)  
hex\_numbers = [num.lstrip(**'0'**) **for** num **in** s1.split()]  
max\_len = len(max(hex\_numbers, key=len))  
print(max\_len)

Вариант 3

**import** re  
  
**with** open(**'24data/24-264.txt'**) **as** f:  
 s = f.read().strip()  
  
pattern = **r'[1-9A-F][0-9A-F]+'**max\_seq = max(re.findall(pattern, s), key=len)  
print(len(max\_seq))

Ответ: 34

2023 БукваЦифра

272) Текстовый файл 24-264.txt состоит не более чем из 106 символов и содержит только заглавные буквы латинского алфавита и цифры. Определите максимальную длину подстроки, в которой ни ода буква не стоит рядом с буквой и ни одна цифра не стоит рядом с цифрой.

Вариант 1

**with** open(**'24data/24-264.txt'**) **as** f:  
 s = f.read().strip()  
max\_len = 0  
cur\_len = 1  
**for** i **in** range(len(s) - 1):  
 **if** s[i].isdigit() != s[i + 1].isdigit():  
 cur\_len += 1  
 **else**:  
 cur\_len = 1  
 max\_len = max(max\_len, cur\_len)  
print(max\_len)

Вариант 2

**from** string **import** ascii\_uppercase, digits  
  
**with** open(**'24data/24-264.txt'**) **as** f:  
 s = f.read().strip()  
  
**for** letter **in** ascii\_uppercase[1:]:  
 s = s.replace(letter, **'A'**)  
**for** digit **in** digits[1:]:  
 s = s.replace(digit, **'0'**)  
**for** i **in** range(2):  
 s = s.replace(**'AA'**, **'A A'**)  
 s = s.replace(**'00'**, **'0 0'**)  
max\_seq = max(s.split(), key=len)  
print(len(max\_seq))

Ответ: 18

2024 CD160

296) Текстовый файл 24-296.txt состоит не более чем из 106 символов и содержит только буквы латинского алфавита A, B, C, D, E и F. Определите максимальное количество идущих подряд символов в прилагаемом файле, среди которых пара символов СD (в указанном порядке) встречается ровно 160 раз.

Вариант 1

**with** open(**'24data/24-296.txt'**) **as** f:  
 s = f.read().strip()  
N = 160  
indexes = [-1]  
**for** i **in** range(len(s) - 1):  
 **if** s[i:i+2] == **'CD'**:  
 indexes.append(i)  
indexes.append(len(s))  
max\_len = 0  
**for** i **in** range(N + 1, len(indexes)):  
 max\_len = max(max\_len, indexes[i] - indexes[i - (N + 1)])  
print(max\_len)

Вариант 2

**with** open(**'24data/24-296.txt'**) **as** f:  
 s = f.read().strip()  
N = 160  
a = s.split(**'CD'**)  
first\_len = len(**'CD'**.join(a[:N + 1])) + 1  
last\_len = len(**'CD'**.join(a[-(N + 1):])) + 1  
max\_len = max(first\_len, last\_len)  
**for** i **in** range(1, len(a) - N):  
 cur\_len = len(**'CD'**.join(a[i: i + N + 1])) + 2  
 max\_len = max(max\_len, cur\_len)  
print(max\_len)

Ответ: 1288

2024 ВычУм

298) Текстовый файл 24-298.txt состоит не более чем из 106 символов и содержит только символы, обозначающие знаки «-», «\*», и цифры 0, 7, 8, 9. Определите в прилагаемом файле максимальное количество идущих подряд символов, которые образуют математически правильную последовательность, в которую входят знаки «-» или «\*» и натуральные числа без незначащих нулей.

Вариант 1

**with** open(**'24data/24-298.txt'**) **as** f:  
 s = f.read().strip()  
max\_len = 0  
cur\_len = 0  
is\_number = **False  
for** i **in** range(len(s)):  
 **if** is\_number:  
 **if** s[i] **in '0789'**:  
 cur\_len += 1  
 max\_len = max(max\_len, cur\_len)  
 **else**: *# s[i] in '-\*'* cur\_len += 1  
 is\_number = **False  
 else**:  
 **if** s[i] **in '789'**:  
 cur\_len += 1  
 max\_len = max(max\_len, cur\_len)  
 is\_number = **True  
 else**:  
 cur\_len = 0  
print(max\_len)

Вариант 2

**with** open(**'24data/24-298.txt'**) **as** f:  
 s = f.read().strip()  
s = s.replace(**'-'**, **'\*'**)  
*# s = s.replace('\*0\*', '\*1\*') понадобилось бы, если бы не гарантия натуральности*a = s.split(**'\*\*'**)  
max\_seq = **''  
for** sub\_str **in** a:  
 ss = sub\_str.split(**'\*0'**)  
 max\_seq = max(max\_seq, \*map(**lambda** line: line.lstrip(**'\*0'**), ss), key=len)  
print(len(max\_seq))

Вариант 3

**import** re  
  
**with** open(**'24data/24-298.txt'**) **as** f:  
 s = f.read().strip()  
pattern = **r'(?:[7-9][07-9]\*[\*-])+[7-9][07-9]\*'**print(max(re.findall(pattern, s), key=len))  
print(len(max(re.findall(pattern, s), key=len)))

Ответ: 177

2024 СумУм=0

300) Текстовый файл 24-300.txt состоит не более чем из 106 символов и содержит только десятичные цифры, а также знаки «+» и «\*» (сложения и умножения). Определите максимальное количество символов в непрерывной последовательности, являющейся корректным арифметическим выражением с целыми неотрицательными числами (без знака), значение которого равно нулю. В этом выражении никакие два знака арифметических операций не стоят рядом, порядок действий определяется по правилам математики. В записи чисел отсутствуют незначащие (ведущие) нули. В ответе укажите количество символов в найденном выражении.

**import** re  
  
**with** open(**'24data/24-300.txt'**) **as** f:  
 s = f.read().strip()  
  
left\_p = **r'(?:[1-9]\d\*\\*|0\\*)\*'**rigth\_p = **r'(?:\\*[1-9]\d\*|\\*0)\*'**term = **fr'{**left\_p**}0{**rigth\_p**}'**pattern = **fr'(?:{**term**}\+)\*{**term**}'**

max\_str = max(re.findall(pattern, s), key=len)  
print(max\_str)  
print(len(max\_str))

Ответ: 126