

Олимпиада «Ломоносов» по информатике  
2023-2024 учебный год. Заключительный тур  
Работа участника с id заявки 1265129, логином inf24f\_267

Сводный итог по всем задачам в проверяющей системе

Run ID	Time	User name	Problem	Language	Result	Tests	Score
68	0:31:48	inf24f_267	1	python3	OK	28	100
157	1:04:09	inf24f_267	2	python3	OK	28	100
601	3:45:31	inf24f_267	3	python3	Partial solution	27	96
429	2:51:54	inf24f_267	5	python3	OK	22	100
396 технических баллов							
66 итоговых баллов							

## Посылка по задаче 1

```
[1] def get_tetra(length=30):
[2]     t = [0] * length
[3]     t[3] = 1
[4]     for i in range(4, length):
[5]         t[i] = t[i - 1] + t[i - 2] + t[i - 3] + t[i - 4]
[6]     return t
[7]
[8]
[9] def is_search(a, t):
[10]     k1 = 0
[11]     ans = ""
[12]     for i in range(len(t) - 1, 3, -1):
[13]         if a >= t[i]:
[14]             # ans += "1"
[15]             k1 += 1
[16]             a -= t[i]
[17]         else:
[18]             pass
[19]             # ans += "0"
[20]     return k1 % 2 == 0
[21]
[22]
[23] t = get_tetra()
[24] n = int(input())
[25] m = 0
[26] for _ in range(n):
[27]     if is_search(int(input()), t):
[28]         m += 1
[29] print(m)
```

---

## Посылка по задаче 2

```
[1] import fractions as frct
[2] import sys
[3]
[4] sys.setrecursionlimit(10000)
[5]
[6]
[7] def func(t, d, level):
[8]     global i
[9]     for k in range(8):
[10]         if t[i] == "Q":
[11]             i += 1
[12]             func(t, d, level * 8)
[13]         else:
[14]             try:
[15]                 d[t[i]] += frct.Fraction(1, level)
[16]             except KeyError:
[17]                 pass
[18]             i += 1
[19]
[20]
[21] def convert(f):
[22]     if f == 0:
[23]         print(0.0)
[24]         return
[25]     if f == 1:
[26]         print(1.0)
[27]         return
[28]     print("0.", end="")
[29]     level = 2
[30]     while f != 0:
[31]         if f >= frct.Fraction(1, level):
[32]             print(1, end="")
[33]             f -= frct.Fraction(1, level)
[34]         else:
[35]             print(0, end="")
[36]             level *= 2
[37]
[38]
[39] t = input()
[40] d = {"W": frct.Fraction(0),
[41]      "R": frct.Fraction(0),
[42]      "O": frct.Fraction(0),
[43]      "Y": frct.Fraction(0),
[44]      "G": frct.Fraction(0),
[45]      "C": frct.Fraction(0),
[46]      "B": frct.Fraction(0),
[47]      "V": frct.Fraction(0),
[48]      "D": frct.Fraction(0)
[49]      }
[50] if t[0] != "Q":
[51]     print(t)
[52]     print(1.0)
[53] else:
[54]     i = 1
[55]     func(t, d, 8)
[56]     #print(d)
[57]     lst = []
[58]     for k, v in d.items():
[59]         lst.append((k, v))
[60]     lst.sort(key=lambda x: (x[1], -ord(x[0])))
[61]     #print(lst)
[62]     print(lst[-1][0])
[63]     convert(lst[-1][1])
```

### Посылка по задаче 3

```
[1] from math import inf
[2]
[3]
[4] def e2dec(s):
[5]     if s == "[]":
[6]         return 0
[7]     ans = 0
[8]     level = 1
[9]     i = len(s)
[10]    k = 1
[11]    a = []
[12]    while i > 0:
[13]        # print(i)
[14]        while True:
[15]            try:
[16]                digit = d[s[i - k:i]]
[17]                break
[18]            except:
[19]                k += 1
[20]        a.append(digit)
[21]        i = i - k
[22]        k = 1
[23]    for i in range(len(a) - 1, -1, -1):
[24]        ans += a[i] * level
[25]        level *= 12
[26]    return ans
[27]
[28]
[29] def dec2e(x):
[30]     if x == 0:
[31]         return "[]"
[32]     if x == 12:
[33]         return "j))"
[34]     ans = []
[35]     while x != 0:
[36]         ans.append(x % 12)
[37]         x //= 12
[38]
[39]     flag = False
[40]     for i in range(len(ans)):
[41]         if ans[i] == 0:
[42]             if not flag:
[43]                 ans[i] = 12
[44]                 flag = True
[45]             else:
[46]                 ans[i] = 11
[47]         else:
[48]             if flag:
[49]                 ans[i] -= 1
[50]                 flag = False
[51]     return "".join(q[ans[i]] for i in range(len(ans)))
[52]
[53]
```

```

[54] d = {"[]": 0,
[55]      "i": 1,
[56]      "i(": 2,
[57]      "i(("": 3,
[58]      "I": 4,
[59]      "I(": 5,
[60]      "I(("": 6,
[61]      "J": 7,
[62]      "J)": 8,
[63]      "J))": 9,
[64]      "j": 10,
[65]      "j)": 11,
[66]      "j))": 12,
[67]      }
[68]
[69] q = ["[]", "i", "i(", "i(("", "I", "I(", "I(("", "J", "J)", "J))", "j", "j)", "j))"]
[70]
[71] n = int(input())
[72] lst = []
[73] for i in range(n):
[74]     s = input()
[75]     lst.append([i + 1, e2dec(s)])
[76]
[77] min_v = -1
[78] min_i = None
[79] max_v = inf
[80] max_i = None
[81]
[82] for i in range(len(lst)):
[83]     if min_i is None:
[84]         min_v = lst[i][1]
[85]         min_i = lst[i][0]
[86]     else:
[87]         if lst[i][1] < min_v:
[88]             min_v = lst[i][1]
[89]             min_i = lst[i][0]
[90]         elif lst[i][1] == min_v:
[91]             min_i = max(min_i, lst[i][0])
[92]     if max_i is None:
[93]         max_v = lst[i][1]
[94]         max_i = lst[i][0]
[95]     else:
[96]         if lst[i][1] > max_v:
[97]             max_v = lst[i][1]
[98]             max_i = lst[i][0]
[99]         elif lst[i][1] == max_v:
[100]            max_i = max(max_i, lst[i][0])
[101] if max_v == min_v:
[102]     min_i -= 1
[103] if min_i > max_i:
[104]     min_i, max_i = max_i, min_i
[105] print(dec2e(min_i))
[106] print(dec2e(max_i))

```

Посылоч по задаче 4 не было

## Посылка по задаче 5

```
[1] #print(chr(33), chr(126))
[2]
[3]
[4] def is_norm(a, b):
[5]     for i in range(33, len(a)):
[6]         if a[i] < b[i]:
[7]             return False
[8]     return True
[9]
[10]
[11] s1 = input()
[12] s2 = input()
[13]
[14] a = [0] * (126)
[15] b = [0] * (126)
[16] for e in s2:
[17]     b[ord(e)] += 1
[18]
[19] p1 = 0
[20] p2 = 0
[21] a[ord(s1[0])] += 1
[22] min_l = len(s1) + 1
[23] ans = None
[24] while p1 < len(s1) and p2 < len(s1) and p1 <= p2:
[25]     while not is_norm(a, b):
[26]         p2 += 1
[27]         if p2 == len(s1):
[28]             break
[29]         a[ord(s1[p2])] += 1
[30]     else:
[31]         if (p2 - p1 + 1) < min_l and p1 <= p2:
[32]             min_l = p2 - p1 + 1
[33]             ans = (p1, p2)
[34]             a[ord(s1[p1])] -= 1
[35]             p1 += 1
[36]
[37] if ans is None:
[38]     print()
[39] else:
[40]     print(s1[ans[0]:ans[1] + 1])
```

Посылоч по задаче 6 не было