**Олимпиада «Ломоносов» по информатике**
**2024-2025 учебный год. Заключительный этап**
**Работа участника с id заявки 1362478, логином inf25f_128**

**Сводный итог по всем задачам в проверяющей системе**

| RunID | Time | Username | Prob | Lang | Result | Tests | Score |
|---|---|---|---|---|---|---|---|
| 156 | 2:47:04 | inf25f_128 | 4 | pypy3 | Partial solution | 1 | 0 |
| 136 | 2:28:06 | inf25f_128 | 3 | pypy3 | Partial solution | 19 | 80 |
| 105 | 1:59:26 | inf25f_128 | 5 | pypy3 | Partial solution | 2 | 10 |
| 49 | 0:56:49 | inf25f_128 | 1 | pypy3 | OK | 29 | 100 |
| 28 | 0:39:57 | inf25f_128 | 2 | pypy3 | OK | 28 | 100 |

290 технических баллов
83 итоговых балла

## Посылка по задаче 1

```
[1]  n1 = int(input())
[2]  n2 = int(input())
[3]  x = 0
[4]  n = n1
[5]  def f(x , memo):
[6]      if x == 1:
[7]          memo.append(4)
[8]          return memo
[9]      if x % 2 == 0:
[10]         memo.append(x//2)
[11]         return f(x//2, memo)
[12]     memo.append(3* x + 1)
[13]     return f(3 * x + 1, memo)
[14] l1 = f(n1,[n1])
[15] l2 = f(n2,[n2])
[16] max = 10e10
[17] for i in range(len(l1)):
[18]     if l1[i] in l2:
[19]         if i + l2.index(l1[i]) <max:
[20]             #print(l1[i], )
[21]             max = i + l2.index(l1[i])
[22] print(max)
```

### Посылка по задаче 2

```
[1]
[2] def convert_to_norm(s):
[3]     k = len(s)
[4]     all_simb = 0
[5]     for i in range(len(s)):
[6]         if s[i] != '^' and s[i] != '~' and s[i] != '_':
[7]             all_simb += 1
[8]     simb = 0
[9]     sum = 0
[10]    for i in range(len(s)):
[11]        if s[i] != '^' and s[i] != '~' and s[i] != '_':
[12]            simb += 1
[13]            num  = ord(s[i]) - 97
[14]            if i + 1 < k:
[15]                if s[i + 1] == '^':
[16]                    num += 26
[17]                if s[i + 1] == '~':
[18]                    num += 52
[19]                if s[i + 1] == '_':
[20]                    num += 78
[21]            m = 26 ** (all_simb - simb)
[22]            sum += m * num
[23]    return  sum
[24]#print(convert_to_norm('y~x^'))
[25]n = int(input())
[26]l = []
[27]for i in range(n):
[28]    s = input()
[29]    l.append(convert_to_norm(s))
[30]l1 = sorted(l)
[31]
[32]indexes = []
[33]for i in range(n):
[34]    if l[i] != l1[i]:
[35]        indexes.append(i)
[36]print(indexes[0] + 1, indexes[1] + 1)
```

### Посылка по задаче 3

```
[1] n , t = map(int,input().split())
[2] tmax = t
[3] p = []
[4] all_s = []
[5] l = []
[6] def obh(i , t, b, ans2):
[7]     global all_s
[8]     ans = []
[9]     for itt in ans2:
[10]        ans.append(itt)
[11]    if i not in ans:
[12]        b += p[i]
[13]    ans.append(i)
[14]
[15]    if tmax < t:
[16]        return
[17]    #print(b, t, ans)
[18]    if b == all_s[i][0]:
[19]        if ans in all_s[i][1]:
[20]            return
[21]        all_s[i][1].append(ans)
[22]    if b >  all_s[i][0]:
[23]        all_s[i][0] = b
[24]        all_s[i][1] = []
[25]        #ans.append(i)
[26]        all_s[i][1].append(ans)
[27]    for it in range(n):
[28]        if it != i:
[29]            obh(it, t + l[i][it], b, ans)
[30]    return
[31] p = list(map(int,input().split()))
[32] for i in range(n):
[33]    all_s.append([0,[]])
[34]    l.append(list(map(int,input().split())))
[35] obh(0,0, 0,[])
[36] max = 0
[37] maxi = 0
[38] answers = all_s[0][1]
[39]
[40] #print(answers, all_s)
[41] an = []
[42] for elm in answers:
[43]    an.append(list(set(elm)))
[44]
[45] min_len = []
[46] #an = [[1, 2, 3], [0, 1, 2], [0, 1, 2, 3, 4]]
[47] min_l = len(an[0])
[48] for i in an:
[49]    if len(i) < min_l:
[50]        min_l = len(i)
[51]        min_len = [i]
[52]    else:
[53]        min_len.append(i)
[54]
[55] answer = sorted(sorted(min_len)[0])
[56] print(len(answer))
[57] for i in answer:
[58]    print(i  + 1,end = " ")
```

## Посылка по задаче 4

```
[1] x1 = input()
[2] print("2H2+O2=2H2O")
[3] x2 = input()
[4] print("Si+O2=SiO2")
[5] x3 = input()
[6] print("C+O2=CO2")
[7] x4 = input()
[8] print("(He[C2(O2Mg)4(O2Fe)2]2N3)2H3+4F=(He(C2(O2Mg)4(O2Fe)2)2N3)2H3F4")
```

### Посылка по задаче 5

```
[1] m, n , k = map(int,input().split())
[2] v = dict()
[3]
[4]
[5] def dfs(i, q, l, last):
[6]     if i == q:
[7]         return l
[8]     else:
[9]         if i in v:
[10]            elms = list(v[i].keys())
[11]            #print(elms)
[12]            for elm in elms:
[13]                if elm != last:
[14]                    m1 = dfs(elm, q, l + [v[i][elm]], i)
[15]                    if m1 != [-1]:
[16]                        return m1
[17]        return [-1]
[18]
[19]
[20]
[21]for i in range(n-1):
[22]    v1, v2, dist = map(int,input().split())
[23]    if v1 in v:
[24]        v[v1][v2] = dist
[25]    else:
[26]        v[v1] = dict()
[27]        v[v1][v2] = dist
[28]    if v2 in v:
[29]        v[v2][v1] = dist
[30]    else:
[31]        v[v2] = dict()
[32]        v[v2][v1] = dist
[33]
[34]for i in range(k):
[35]    p, q = map(int,input().split())
[36]    pos_ch = sorted(list(set(dfs(p,q, [],-1))))
[37]    max = 0
[38]    last = -1
[39]    for i in range(len(pos_ch)):
[40]        if pos_ch[i] - last  - 1 > max:
[41]            max =  pos_ch[i] - last -1
[42]        last = pos_ch[i]
[43]    print(max)
```