Олимпиада «Ломоносов» по информатике
2024-2025 учебный год. Заключительный этап
Работа участника с id заявки 1397166, логином inf25f_321

Сводный итог по всем задачам в проверяющей системе

RunID	Time	Username	Prob	Lang	Result	Tests	Score
207	3:01:26	inf25f_321	4	pypy3	Partial solution	1	0
202	2:59:04	inf25f_321	5	pypy3	Partial solution	1	0
159	2:21:55	inf25f_321	3	pypy3	OK	23	100
32	0:42:01	inf25f_321	2	python3	OK	28	100
2	0:22:47	inf25f_321	1	python3	Partial solution	12	32

232 технических балла
66 итоговых баллов

Посылка по задаче 1

```
[1] import heapq, collections
[2]
[3] def main():
[4]     n1 = int(input())
[5]     n2 = int(input())
[6]     if n1 == n2:
[7]         print(0)
[8]         return
[9]     Res = collections.defaultdict(tuple)
[10]    Res[n1] = (n1,)
[11]    heap = [(0, n1)]
[12]    def f(x):
[13]        if x % 2 == 0:
[14]            return x // 2
[15]        return x * 3 + 1
[16]    while heap:
[17]        path, v = heapq.heappop(heap)
[18]        if (v - 1) % 3 == 0 and (v - 1) // 3 not in Res:
[19]            heapq.heappush(heap, (path + 1, (v - 1) // 3))
[20]            Res[(v - 1) // 3] = Res[v] + ((v - 1) // 3,)
[21]            if (v - 1) // 3 == n2:
[22]                break
[23]        if v * 2 not in Res:
[24]            heapq.heappush(heap, (path + 1, v * 2))
[25]            Res[v * 2] = Res[v] + (v * 2,)
[26]            if v * 2 == n2:
[27]                break
[28]        x = f(v)
[29]        if x not in Res:
[30]            heapq.heappush(heap, (path + 1, x))
[31]            Res[x] = Res[v] + (x,)
[32]            if x == n2:
[33]                break
[34]    print(len(Res[n2])-1)
[35]    if len(Res[n2])-1 > 1:
[36]        print(*Res[n2][1:-1])
[37]
[38] main()
```

Посылка по задаче 2

```
[1] import re
[2]
[3] def convert(s):
[4]     res = 0
[5]     B = sorted('qwertyuiopasdfghjklzxcvbnmQWERTYUIOPASDFGHJKLZXCVBNM')
[6]     C = {B[i]: i for i in range(len(B))}
[7]     A = re.findall('[a-zA-Z][_^~]*', s)[::-1]
[8]     for i in range(len(A)):
[9]         x = C[A[i][0]]
[10]        if len(A[i]) == 2:
[11]            x += 52 * {'^': 1, '~': 2, '_': 3}[A[i][1]]
[12]        res += x * 52 ** i
[13]    return res
[14]
[15]
[16] def main():
[17]    n = int(input())
[18]    A = [(convert(input()), i) for i in range(n)]
[19]    B = sorted(A, key=lambda a: (-a[0], a[1]))
[20]    R = []
[21]    for i in range(len(A)):
[22]        if B[i][1] != i:
[23]            R.append(i)
[24]    R.sort()
[25]    print(R[0] + 1, R[1] + 1)
[26]
[27] main()
```

Посылка по задаче 3

```
[1]  def func1(List, P, Q, t):
[2]      mxres = 0
[3]      A = []
[4]      for i in List:
[5]          s = bin(i)[2:].rjust(len(P)-1, '0')
[6]          B = {i+1 for i in range(len(s)) if s[i] == '1'}
[7]          temp = rec1(B, P, t)
[8]          if temp:
[9]              res = sum(Q[i] for i in temp)
[10]             if res > mxres:
[11]                 A = []
[12]                 A.append(sorted(temp))
[13]                 mxres = res
[14]             elif res == mxres:
[15]                 A.append(sorted(temp))
[16]     if not A:
[17]         return Q[0], (0,)
[18]     return mxres, sorted(A)[0]
[19]
[20] def rec1(B: set, P: list[list[int]], t: int, t_now: int=0, path: tuple=(0,)):
[21]     if t_now + P[path[-1]][0] > t:
[22]         return False
[23]     if not B:
[24]         return path
[25]     for i in B.copy():
[26]         B.remove(i)
[27]         temp = rec1(B, P, t, t_now + P[path[-1]][i], path + (i,))
[28]         if temp:
[29]             return temp
[30]         B.add(i)
[31]     return False
[32]
[33] def func2():
[34]     pass
[35]
[36] def main():
[37]     n, t, m = map(int, input().split())
[38]     Q = list(map(int, input().split()))
[39]     P = [list(map(int, input().split())) for i in range(n)]
[40]     for i in range(m):
[41]         u, v, d = map(int, input().split())
[42]         P[u-1][v-1] = min(d, P[u-1][v-1])
[43]         P[v-1][u-1] = min(d, P[v-1][u-1])
[44]     for u in range(n):
[45]         for v in range(n):
[46]             if u != v:
[47]                 for k in range(n):
[48]                     if k != u and k != v:
[49]                         P[u][v] = min(P[u][v], P[u][k] + P[k][v])
[50]     C = {i: [] for i in range(1, len(P) + 1)}
[51]     for i in range(1, 2**(len(P)-1)):
[52]         C[bin(i).count('1')].append(i)
[53]     mxres = Q[0]
[54]     respath = (0,)
[55]     for i in range(1, len(P) + 1):
[56]         #timer = time.time()
[57]         res = func1(C[i], P, Q, t)
[58]         #timer2 = time.time() - timer
[59]         #print(i, res, timer2)
[60]         if res[0] > mxres:
[61]             mxres = res[0]
[62]             respath = res[1]
[63]         elif res[0] == mxres and res[1] < respath:
[64]             respath = res[1]
[65]         if res[1] == (0,):
[66]             break
[67]     print(len(respath))
[68]     print(*[i+1 for i in respath])
[69]
[70] #timer = time.time()
[71]
[72] main()
```

Посылка по задаче 4

```
[1] print('2H2+O2=2H2O')
[2] print('Si+O2=SiO2')
[3] print('C+O2=CO2')
[4] print('(He[C2(O2Mg)4(O2Fe)2]2N3)2H3+4F=(He(C2(O2Mg)4(O2Fe)2)2N3)2H3F4')
```

Посылка по задаче 5

```
[1] import collections
[2]
[3] def cmp1(m, a, b):
[4]     mxres = 0
[5]     a, b = bin(a)[2:], bin(b)[2:]
[6]     countx = 0
[7]     for i in range(1, len(a)):
[8]         if a[i] == b[i] == '1':
[9]             countx += 1
[10]        else:
[11]            mxres = max(mxres, countx)
[12]            countx = 0
[13]    return mxres
[14]
[15]def main():
[16]    m, n, k = map(int, input().split())
[17]    A = [2**(m+1)-1 for _ in range(n)]
[18]    B = collections.defaultdict(list)
[19]    for i in range(n - 1):
[20]        s, d, b = map(int, input().split())
[21]        B[s-1].append((d-1, b+1))
[22]    color = [False for _ in range(n)]
[23]    Queue = collections.deque([0])
[24]    while Queue:
[25]        a: int = Queue.pop()
[26]        if color[a]:
[27]            continue
[28]        color[a] = True
[29]        for i in B[a]:
[30]            A[i[0]] = A[a] - 2**(m-i[1])
[31]            Queue.appendleft(i[0])
[32]    for _ in range(k):
[33]        p, q = map(int, input().split())
[34]        a, b = A[p-1], A[q-1]
[35]        print(cmp1(m, a, b))
[36]
[37]main()
```