**20743:整人的提词本**

def cs20743():  
 ipt = input()  
 lst = []  
 ans = []  
 for i in ipt:  
 if i == ")":  
 p = lst.pop()  
 while p != "(":  
 ans.append(p)  
 p = lst.pop()  
 lst += ans  
 ans = []  
 else:  
 lst.append(i)  
 ans = ""  
 for i in lst:  
 ans += i  
 print(ans)



## 02255:重建二叉树

def cs02255():  
 try:  
 while True:  
 fore, mid = input().split()  
 ans = [""]  
  
 def fm(f, m):  
 i = m.index(f[0])  
 if i >= 1:  
 fm(f[1:i + 1], m[:i])  
 if i + 1 < len(f):  
 fm(f[i + 1:], m[i + 1:])  
 ans[0] += f[0]  
  
 fm(fore, mid)  
 print(ans[0])  
 except:  
 exit()



## 01426:Find The Multiple

def cs01426():  
 flag = [True]  
  
 def s(k, tmp, n):  
 if not n:  
 exit()  
 if flag[0]:  
 if k <= 99:  
 for i in ["0", "1"]:  
 s(k + 1, tmp + i, n)  
 else:  
 tmp = int(tmp)  
 if not tmp % n and tmp:  
 print(tmp)  
 flag[0] = False  
  
 while True:  
 s(0, "", int(input()))  
 flag = [True]



## 04115:鸣人和佐助

def cs04115():  
 m, n, c0 = map(int, input().split())  
 lst, clst = [], []  
 itm = [(0, 1), (0, -1), (1, 0), (-1, 0)]  
 for i in range(m):  
 ipt = input()  
 lst.append([j for j in ipt])  
 clst.append([c0 for j in range(n)])  
 f = ipt.find("@")  
 if f + 1:  
 x0, y0 = i, f  
 t, xy = 0, [(x0, y0)]  
 while xy:  
 t += 1  
 xyi = []  
 while xy:  
 j = xy.pop()  
 lst[j[0]][j[1]] = False  
 for i in itm:  
 xi, yi = j[0] + i[0], j[1] + i[1]  
 if 0 <= xi < m and 0 <= yi < n and lst[xi][yi]:  
 if (xi, yi) not in xyi:  
 if lst[xi][yi] == "\*":  
 xyi.append((xi, yi))  
 clst[xi][yi] = clst[j[0]][j[1]]  
 elif lst[xi][yi] == "#" and clst[j[0]][j[1]] > 0:  
 xyi.append((xi, yi))  
 clst[xi][yi] = clst[j[0]][j[1]] - 1  
 elif lst[xi][yi] == "+":  
 print(t)  
 exit()  
 else:  
 if lst[xi][yi] == "\*":  
 clst[xi][yi] = max(clst[j[0]][j[1]], clst[xi][yi])  
 elif lst[xi][yi] == "#":  
 clst[xi][yi] = max(clst[j[0]][j[1]] - 1, clst[xi][yi])  
  
 xy = xyi  
 print(-1)



## 20106:走山路

def cs20106():  
 m, n, p = map(int, input().split())  
 lst = []  
 for i in range(m):  
 lst.append(input().split())  
 for i in range(p):  
 ax0, ay0, bx0, by0 = map(int, input().split())  
 if lst[ax0][ay0] == "#" or lst[ax0][ay0] == "#":  
 print("NO")  
 else:  
 ans = [10000]  
 flst = [[-1 for j in range(n)] for jj in range(m)]  
 flst[ax0][ay0] = 0  
 xy = [(ax0, ay0)]  
 while xy:  
 xyi = []  
 while xy:  
 ax, ay = xy.pop()  
 if ax == bx0 and ay == by0:  
 ans[0] = min(ans[0], flst[ax][ay])  
 else:  
 for j in [(1, 0), (-1, 0), (0, 1), (0, -1)]:  
 ax1, ay1 = ax + j[0], ay + j[1]  
 if 0 <= ax1 < m and 0 <= ay1 < n and lst[ax1][ay1] != "#":  
 tmp = flst[ax][ay] + abs(int(lst[ax1][ay1]) - int(lst[ax][ay]))  
 if flst[ax1][ay1] == -1 or flst[ax1][ay1] > tmp:  
 flst[ax1][ay1] = tmp  
 if (ax1, ay1) not in xyi:  
 xyi.append((ax1, ay1))  
 xy = xyi  
 if ans[0] == 10000:  
 print("NO")  
 else:  
 print(ans[0])



## 05442:兔子与星空

def cs05442():  
 dct = {}  
 n = int(input()) - 1  
 for i in range(n):  
 ipt = input().split()  
 dct[ipt[0]] = []  
 for j in range(int(ipt[1])):  
 dct[ipt[0]].append((ipt[j \* 2 + 2], int(ipt[j \* 2 + 3])))  
 s = "ABCDEFGHIJKLMNOPQRSTUVWXYZa"  
 tmp = []  
 for i in range(n):  
 tmp += [(i, s.find(j[0]), j[1]) for j in dct[s[i]]]  
 tmp.sort(key=lambda i: i[2])  
 dest = [-1 for i in range(n + 1)]  
 ans, j = 0, 0  
 for i in tmp:  
 iid, jid = i[0], i[1]  
 while dest[iid] + 1:  
 iid = dest[iid]  
 while dest[jid] + 1:  
 jid = dest[jid]  
 if iid-jid:  
 dest[iid] = jid  
 ans += i[2]  
 j += 1  
 if j == n:  
 break  
  
 print(ans)  
  
  
cs05442()



总结：我花了很多时间，原因之一是思维定势看到题就想着用dfs做，然后意识到才改过来。

鸣人这题我卡了很久，最终Ac也是学到了很多减枝的办法。每题我都看了其他同学的运行时间，学了很多简洁的思路