#狄克斯特拉算法

'''

参数：S，T，B

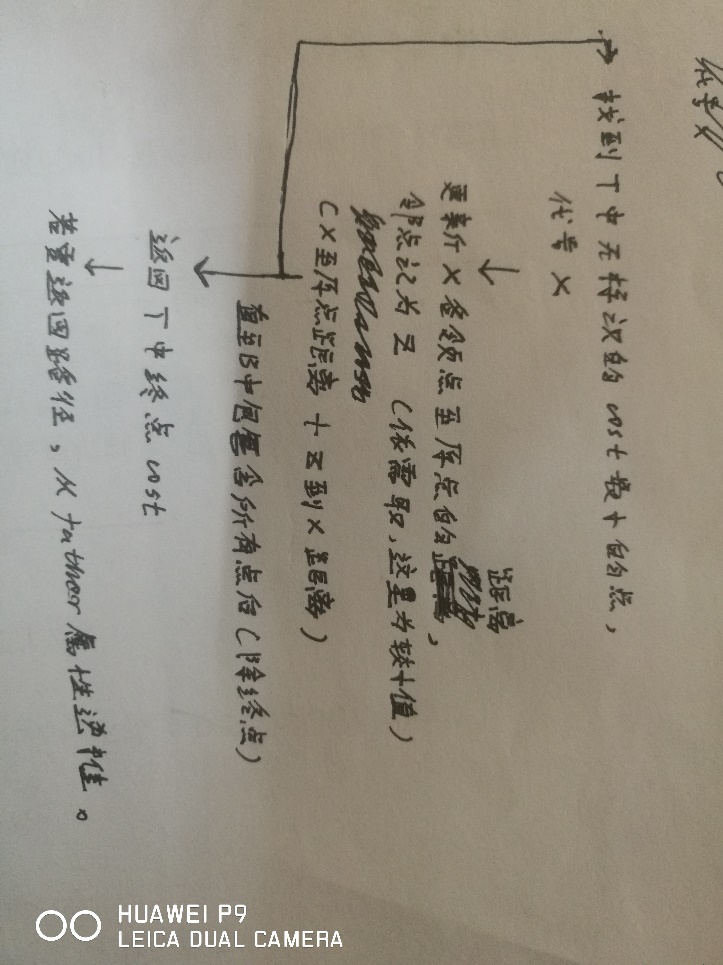
B为空列表，用来储存以标识点

S为各点的邻点，以及距离邻点的距离（终点没有邻点，故不记录）

T为目前已知的各点距离远点的最短路径距离，以及父点

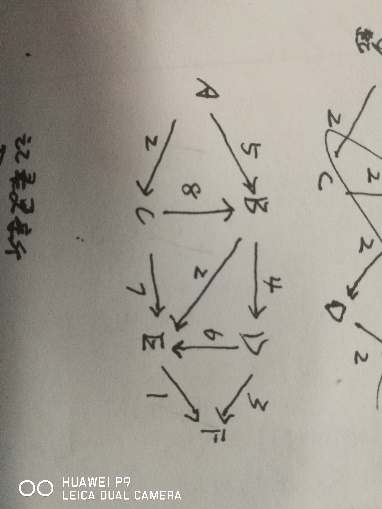
注：float('inf')为无穷大

思路：



'''

例：



def dikeSTL (S,T,B):

if len(B)==len(S):

return T['F']['cost']

x = min([i for i in S if i not in B],key=lambda x:T[x]['cost'])

for i in S[x]:

new\_cost = S[x][i]+T[x]['cost']

if new\_cost<T[i]['cost']:

T[i]['cost']=new\_cost

T[i]['father']=x

B.append(x)

return dikeSTL(S,T,B)

S={'A':{'B':5,'C':2},'B':{'D':4,'E':2},'C':{'B':8,'E':7},'D':{'E':6,'F':3},'E':{'F':1}}

T={'A':{'father':'A','cost':0},'B':{'father':'','cost':float('inf')},'C':{'father':'','cost':float('inf')},'D':{'father':'','cost':float('inf')},

'E':{'father':'','cost':float('inf')},'F':{'father':'','cost':float('inf')}}

B=[]

dikeSTL(S,T,B)