Ağ Optimizasyonu



Ad : Ege Erbilen

Okul No : 160255080

Bölümü : Bilgisayar Mühendisliği

Sınıf : 4.sınıf

Ağ Optimizasyonu Dersi – Prim Algoritması

**Çalışma Adı:** Reklam panoları(Dijital reklam panoları) arasında en kısa yoldan giderek panolar arası veri gönderimini sağlamak.

**Problemin Tanımı:** Bu problemde alışveriş merkezleri içerisinde belirli noktalara dijital reklam panoları konacaktır. Reklam panolarında aynı anda aynı görseller gösterilmelidir aynı zamanda da bu reklam panolarına yeni reklam yüklenmesi tek tek yükleme yapmak yerine sistemin bir merkezi olması ve o merkezden reklamların değiştirilmesi gerekiyor. Alışveriş merkezi içerisindeki reklam panoları arasına birbirleri ile bağlantılı olacak şekilde kablo çekilmesi gerekiyor ki bu sayede tek bir merkezden rahat bir şekilde kontrol edile bilsin. Cihazlar arası kabloların optimal şekilde bir biri ile bağlanması istenmektedir.

**Alışveriş merkezi içerisindeki reklam panolarının bir birlerine olan uzaklıkları**

Kontrol Merkezi

980m

560m 605m

540m

345m 750m

320m

450m 425m 240m

380m

650m

150m 195m

120m 780m

160m

360m 265m

260m

110m

490m 420m

Yukarı tarafta görmüş olduğunuz problem kapsamında kullanılan reklam panoları arasında gidilebilecek olan yollar gösterilmektedir. Reklam panoları arası uzaklıklar şekilde rahatça görünmektedir. Bu yollar arasında reklam panolarına kablolar ile veri gönderilecektir bunun için en kısa yolu bulmamız gerekir bu yolu bulmak için ise Prim algoritmasından faydalanırız ve en kısa yolu buluruz. Sıfır noktası panoların kontrol merkezidir.

**Reklam Panoları Arası Uzaklık Matrisi**

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | **0** | **1** | **2** | **3** | **4** | **5** | **6** | **7** | **8** | **9** | **10** | **11** | **12** |
| **0** | **-** | **540** | **560** | **980** | **-** | **-** | **-** | **-** | **-** | **-** | **-** | **-** | **-** |
| **1** | **540** | **-** | **-** | **-** | **-** | **450** | **-** | **380** | **-** | **-** | **-** | **-** | **-** |
| **2** | **560** | **-** | **-** | **605** | **345** | **320** | **-** | **-** | **-** | **-** | **-** | **-** | **-** |
| **3** | **980** | **-** | **605** | **-** | **-** | **-** | **750** | **-** | **-** | **-** | **-** | **-** | **-** |
| **4** | **-** | **345** | **-** | **-** | **-** | **425** | **240** | **-** | **650** | **-** | **-** | **-** | **-** |
| **5** | **450** | **320** | **-** | **-** | **425** | **-** | **-** | **-** | **-** | **150** | **-** | **780** | **-** |
| **6** | **-** | **-** | **-** | **750** | **240** | **-** | **-** | **-** | **195** | **-** | **-** | **-** | **-** |
| **7** | **380** | **-** | **-** | **-** | **-** | **-** | **-** | **-** | **-** | **120** | **360** | **-** | **-** |
| **8** | **-** | **-** | **-** | **-** | **650** | **-** | **195** | **-** | **-** | **-** |  | **265** | **160** |
| **9** | **-** | **-** | **-** | **-** | **-** | **150** | **-** | **120** | **-** | **-** | **110** | **260** | **-** |
| **10** | **-** | **-** | **-** | **-** | **-** | **-** | **-** | **360** | **-** | **110** | **-** | **490** | **-** |
| **11** | **-** | **-** | **-** | **-** | **-** | **780** | **-** | **-** | **265** | **260** | **490** | **-** | **420** |
| **12** | **-** | **-** | **-** | **-** | **-** | **-** | **-** | **-** | **160** | **-** | **-** | **420** | **-** |

Biz bu problemdeki sorunu çözmek için prim algoritmasını kullanırız çünkü dijital reklam panoları arasında ki en kısa yolu bulmamız beklenmektedir. Prim algoritmasını kullanmamızın sebebi ise reklam panoları arası yönsüz ve ağırlıklıdır. Prim algoritması reklam panoları arasındaki düğümleri en az maliyet olacak şekilde bulmamıza olanak sağlayacaktır.

Prim algoritmasının problemimizde işleyişi şu şekildedir öncelikle kontrol merkezine bağlı olan en yakın düğümlere bakarız 0 a en yakın düğüme buluruz ve oraya gideriz gittiğimiz bu düğüm 1 dir ardından 0 ve 1 düğümlerinden birine en yakın bağlı düğüme gider bu düğüm ise 7 dir. Bu seferde ise 0,1 ve 7 düğümlerinden birbirlerine en yakın olan düğüme gideriz bu da 9 dur. 0,1,7 ve 9 düğümlerinden birbirlerine en yakın olan düğüme baktığımızda ise 10 gelir. 0,1,7,9 ve 10 olur bu düğümlere en yakın düğüm 9 dan 5 e dir. Bu şekilde devam edecektir.

Yukarıda gördüğümüz gibi öncelikle düğümün bağlı olduğu en kısa yol bulunur ardından oraya gidilir artık 1. ve 2. düğümlerin bağlı olduğu en kısa yola gidilir ve bu şekilde devam eder sonucunda ise düğümler(dijital reklam panoları) arasındaki en kısa yolu bulur.

**Kullanılan Kod**

Kod python dili ile yazılmıştır

import math

sonsuz = math.inf

düğüm\_sayisi = 13

G = [[0, 540 ,560 ,980 ,0 ,0 ,0 ,0 ,0 ,0 ,0 ,0 ,0],

     [540, 0 ,0 ,0 ,0 ,450 ,0 ,380 ,0 ,0 ,0 ,0 ,0],

     [560, 0 ,0 ,605 ,345 ,320 ,0 ,0 ,0 ,0 ,0 ,0 ,0],

     [980, 0 ,605 ,0 ,0 ,0 ,750 ,0 ,0 ,0 ,0 ,0 ,0],

     [0, 345 ,0 ,0 ,0 ,425 ,240 ,0 ,650 ,0 ,0 ,0 ,0],

     [450, 320 ,0 ,0 ,425 ,0 ,0 ,0 ,0 ,150 ,0 ,780 ,0],

     [0, 0 ,0 ,750 ,240 ,0 ,0 ,0 ,195 ,0 ,0 ,0 ,0],

     [380, 0 ,0 ,0 ,0 ,0 ,0 ,0 ,0 ,120 ,360 ,0 ,0],

     [0, 0 ,0 ,0 ,650 ,0 ,195 ,0 ,0 ,0 ,0 ,265 ,160],

     [0, 0 ,0 ,0 ,0 ,150 ,0 ,120 ,0 ,0 ,110 ,260 ,0],

     [0, 0 ,0 ,0 ,0 ,0 ,0 ,360 ,0 ,110 ,0 ,490 ,0],

     [0, 0 ,0 ,0 ,0 ,780 ,0 ,0 ,265 ,260 ,490 ,0 ,420],

     [0, 0 ,0 ,0 ,0 ,0 ,0 ,0 ,160 ,0 ,0 ,420 ,0]

     ]

print("Matris: ")

for i in G:

    print(i)

print()

print()

s\_matrisi = ""

mesafe = [0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0]

baslangic = 0

mesafe[0] = True

topla = 0

print("Kullanılan düğümler ve mesafeler: " )

while (baslangic < düğüm\_sayisi - 1):

    minimum = sonsuz

    x = 0

    y = 0

    for i in range(düğüm\_sayisi):

        if mesafe[i]:

            for j in range(düğüm\_sayisi):

                if ((not mesafe[j]) and G[i][j]):

                    if minimum > G[i][j]:

                        minimum = G[i][j]

                        x = i

                        y = j

    s\_matrisi = s\_matrisi + str(x) + "-" + str(y) + " "

    print(str(x) + "-" + str(y) + ":" + str(G[x][y]))

    topla = topla + G[x][y]

    mesafe[y] = True

    baslangic += 1

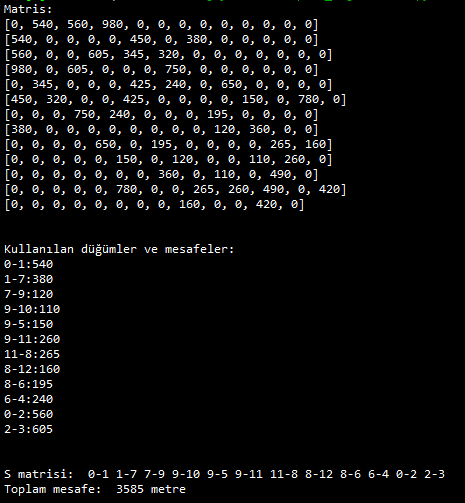
print()

print()

print("S matrisi: ", s\_matrisi)

print("Toplam mesafe: ",topla,"metre")

**Ekran Çıktısı**



**Sonuç**

Kontrol Merkezi

980m

560m 605m

540m

345m 750m

320m

450m 425m 240m

380m

650m

150m 195m

120m 780m

160m

360m 265m

260m

110m

490m 420m

**Kullanılan Düğümler ve Mesafeleri**

0-1:540

1-7:380

7-9:120

9-10:110

9-5:150

9-11:260

11-8:265

8-12:160

8-6:195

6-4:240

0-2:560

2-3:605

**S matrisi:** 0-1, 1-7, 7-9, 9-10, 9-5, 9-11, 11-8, 8-12, 8-6, 6-4, 0-2, 2-3

**Toplam mesafe:** 3585 (metre)