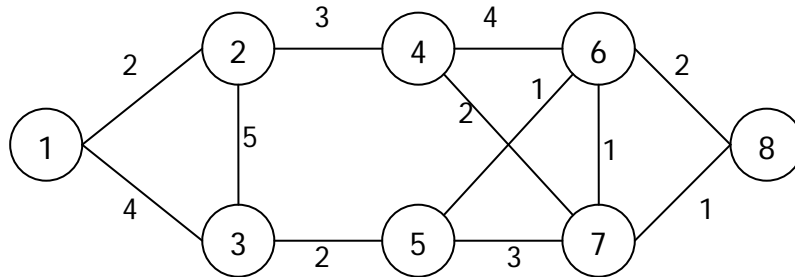


HW4

All Pair Shortest Path Decomposition Algorithm (T.C. Hu, Combinatorial Algorithms)

โจทยัสมมติ



สรุปล

พบว่าการใช้วิธีแบบปกติ Floyd and Warshal นั้นใช้จำนวนการ Run 8 ยกกำลัง 3 รอบ $(n^3) = 512$ รอบ แต่วิธี Decomposition Algorithm สามารถแบ่ง Matrix เพื่อ Run โปรแกรมได้ และใช้จำนวนรอบทั้งสิ้น 5 ยกกำลัง 3 + 5 ยกกำลัง 3 + 5 ยกกำลัง 3 + 2 คูณ 3 คูณ 2 คูณ 3 รอบ $((n_1 + n_2)^3 + (n_2 + n_3)^3 + (n_1 + n_2)^3 + 2(n_1 \cdot n_2 \cdot n_3)) = 411$ รอบ 0

ดังนั้นจึงสรุปได้ว่าวิธี All Pair Shortest Path Decomposition Algorithm มีประสิทธิภาพดีกว่าการทำแบบปกติ

วิธีที่ 1 ทำแบบปกติ (Floyd and Warshal) ไม่มีการ Decomposition

Run โดยโปรแกรม MathLab

```
>> D
```

D =

Columns 1 through 6

1000	2	4	5	6	7
2	1000	5	3	7	6
4	5	1000	6	2	3
5	3	6	1000	4	3
6	7	2	4	1000	1
7	6	3	3	1	1000
7	5	4	2	2	1
8	6	5	3	3	2

Columns 7 through 8

7	8
5	6
4	5
2	3
2	3
1	2
1000	1
1	1000

```
>> R
```

R =

1	2	3	2	3	5	4	7
1	2	3	4	3	7	4	7
1	2	3	7	5	5	6	6
2	2	7	4	7	7	7	7
3	3	3	7	5	6	6	6
5	7	5	7	5	6	7	8
4	4	6	4	6	6	7	8
7	7	6	7	6	6	7	8

Algorithm in MathLab

```
for i=1:8
    for j=1:8
        R(i,j)=j;
    end
end
D = [1000 2 4 1000 1000 1000 1000 1000
     2 1000 5 3 1000 1000 1000 1000
     4 5 1000 1000 2 1000 1000 1000
     1000 3 1000 1000 1000 4 2 1000
     1000 1000 2 1000 1000 1 3 1000
     1000 1000 1000 4 1 1000 1 2
     1000 1000 1000 2 3 1 1000 1
     1000 1000 1000 1000 1000 2 1 1000]
for k=1:8
    for j=1:8
        for i=1:8
            if i ~= j
```

```

        if D(i,j)>D(i,k)+D(k,j)
            D(i,j)=D(i,k)+D(k,j);
            R(i,j)=k;
        end
    end
end
end
end
end
end

```

วิธีที่ 2 ทำแบบปกติ (Floyd and Warshal) มีการ Decomposition

Run โดยโปรแกรม MathLab

>> D

D =

Columns 1 through 6

1000	2	4	5	6	7
2	1000	5	3	7	6
4	5	1000	6	2	3
5	3	6	1000	4	3
6	7	2	4	1000	1
7	6	3	3	1	1000
7	5	4	2	2	1
8	6	5	3	3	2

Columns 7 through 8

7	8
5	6
4	5
2	3
2	3
1	2
1000	1
1	1000

>> R

R =

1	2	3	2	3	5	4	7
1	2	3	4	3	7	4	7
1	2	3	5	5	5	6	7
2	2	5	4	7	7	7	7
3	3	3	7	5	6	6	6
5	7	5	7	5	6	7	8
4	4	6	4	6	6	7	8
7	7	7	7	6	6	7	8

Algorithm in Matlab

```

for i=1:8
    for j=1:8
        R(i,j)=j;
    end
end
D = [1000 2 4 1000 1000 1000 1000 1000
      2 1000 5 3 1000 1000 1000 1000
      4 5 1000 1000 2 1000 1000 1000

```

```

1000 3 1000 1000 1000 4 2 1000
1000 1000 2 1000 1000 1 3 1000
1000 1000 1000 4 1 1000 1 2
1000 1000 1000 2 3 1 1000 1
1000 1000 1000 1000 1000 2 1 1000]
for k=1:5
    for j=1:5
        for i=1:5
            if i ~= j
                if D(i,j)>D(i,k)+D(k,j)
                    D(i,j)=D(i,k)+D(k,j);
                    R(i,j)=k;
                end
            end
        end
    end
end
for k=4:8
    for j=4:8
        for i=4:8
            if i ~= j
                if D(i,j)>D(i,k)+D(k,j)
                    D(i,j)=D(i,k)+D(k,j);
                    R(i,j)=k;
                end
            end
        end
    end
end
for k=1:5
    for j=1:5
        for i=1:5
            if i ~= j
                if D(i,j)>D(i,k)+D(k,j)
                    D(i,j)=D(i,k)+D(k,j);
                    R(i,j)=k;
                end
            end
        end
    end
end
for k=4:5
    for j=6:8
        for i=1:3
            if i ~= j
                if D(i,j)>D(i,k)+D(k,j)
                    D(i,j)=D(i,k)+D(k,j);
                end
            end
        end
    end
end
for k=4:5
    for j=1:3
        for i=6:8
            if i ~= j
                if D(i,j)>D(i,k)+D(k,j)
                    D(i,j)=D(i,k)+D(k,j);
                end
            end
        end
    end
end
for k=4:8

```

```
for j=6:8
  for i=1:3
    if i ~= j
      if D(i,j)==D(i,k)+D(k,j)
        R(i,j)=k;
      end
    end
  end
end
end
end
end
for k=4:8
  for j=1:3
    for i=6:8
      if i ~= j
        if D(i,j)==D(i,k)+D(k,j)
          R(i,j)=k;
        end
      end
    end
  end
end
end
end
```