



ใบงานที่ 13

เรื่อง โครงสร้างข้อมูลกราฟ

เสนอ

อาจารย์ ปิยพล ยืนยงสถาวร

จัดทำโดย

นาย สารินทร์ อินดีะรักษา รหัส 65543206082-1

ใบงานนี้เป็นส่วนหนึ่งของรายวิชา โครงสร้างข้อมูลและขั้นตอนวิธี

หลักสูตรวิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์

มหาวิทยาลัยเทคโนโลยีราชมงคลล้านนา

ประจำภาคที่ 1 ปีการศึกษา 2566

## คำสั่ง/คำชี้แจง

- สร้างโค้ดโปรแกรมตามตัวอย่างในเอกสารประกอบการสอนสัปดาห์ที่7
- แสดงโค้ดโปรแกรมเป็นส่วนๆ พร้อมอธิบาย
- แสดงผลการรันโปรแกรม พร้อมอธิบายการทำงาน
- สรุปผลการทดลอง

## ลำดับขั้นตอนการทดลอง

```
int graph[MaxNode][MaxNode] = {
    {0,1,1,1,0,0},
    {1,0,1,0,1,0},
    {1,1,0,0,0,0},
    {1,0,0,0,1,1},
    {0,1,0,1,0,0},
    {0,0,0,1,0,0}
}; //Declare array and keep data of graph
char NodeName[MaxNode] = {'A','B','C','D','E','F'}; //Keep Node Name
void DispArray2D() //Display value in 2D Array
{
    int i,j; //i=Row, j=Column
    printf(" ");
    for (j=0;j<=MaxNode;j++) //Display column name of array
        printf("%c ",NodeName[j]);
    printf("\n"); //Line feed
    for (i=0;i<MaxNode;i++) //row Loop
    {
        printf("%c ",NodeName[i]); //Display row name of array
        for (j=0;j<MaxNode;j++) //column Loop
            printf("%d ",graph[i][j]); //Display value path
        printf("\n");
    }
}
```

โค้ดส่วนนี้ประกาศและกำหนดค่าให้กับอาร์เรย์ชื่อ "graph" ซึ่งเป็นเมทริกซ์สองมิติขนาด MaxNode x MaxNode โดยใช้ค่า 0 และ 1 เพื่อแสดงความสัมพันธ์ระหว่างโหนด (nodes) ในกราฟ โดยมีกฎเกณฑ์ดังนี้:

- ถ้า graph[i][j] เท่ากับ 1 แสดงว่ามีเส้นเชื่อม (edge) ระหว่างโหนด i และ j
- ถ้า graph[i][j] เท่ากับ 0 แสดงว่าไม่มีเส้นเชื่อมระหว่างโหนด i และ j
- อาร์เรย์ "NodeName" ใช้เก็บชื่อของโหนดแต่ละโหนด เช่น NodeName[0] เก็บค่า 'A', NodeName[1] เก็บค่า 'B', และอื่นๆ ตามลำดับ.
- ในฟังก์ชัน "DispArray2D", มีการใช้ตัวแปร i และ j เพื่อใช้ในการวนลูปเพื่อแสดงข้อมูลในอาร์เรย์ "graph" และชื่อของโหนดในตาราง 2 มิติที่จะแสดงผล.
- การแสดงผลเริ่มต้นด้วยการแสดงชื่อคอลัมน์ของอาร์เรย์ "NodeName" ด้วยการวนลูปของ j และแสดงแต่ละชื่อคอลัมน์ในรูปแบบตาราง.
- หลังจากนั้นในลูป i, จะแสดงชื่อแถวของอาร์เรย์ "NodeName" และแสดงค่าในอาร์เรย์ "graph" ที่ตำแหน่ง (i, j) ด้วยการวนลูปของ j.
- การแสดงผลจะมีการเพิ่มเครื่องหมายเว้นบรรทัดที่ทำยแต่ละแถวเพื่อให้ตารางอ่านง่ายขึ้น.

```

void DispSetOfVertex() //Display set of Vertex
{
    int i;
    printf("\nSet of Vertex = {");
    for (i=0;i<MaxNode;i++)
    {
        printf("%c",NodeName[i]); //Display each node name
        if(i != MaxNode-1)
            printf(",");
    }
    printf("}\n");
}

void DispSetOfEdge() //Display set of Edge
{
    int i,j;
    printf("\nSet of Edge = {");
    for (i=0;i<MaxNode;i++) //row Loop
        for (j=0;j<MaxNode;j++) //column Loop
        {
            if(graph[i][j]==1)
                printf("(%c,%c)",NodeName[i],NodeName[j]); //Show each Edge
        }
    printf("}\n");
}

```

- DispSetOfVertex: ฟังก์ชันนี้ใช้ในการแสดงชุดของโหนด (vertices) ในกราฟ
  - ฟังก์ชันนี้จะแสดงชุดของโหนดในกราฟที่เก็บไว้ในอาร์เรย์ "NodeName"
  - มีการใช้ลูปเพื่อวนลูปผ่านโหนดแต่ละโหนดและแสดงชื่อของโหนดนั้น
- DispSetOfEdge: ฟังก์ชันนี้ใช้ในการแสดงชุดของเส้นเชื่อม (edges) ในกราฟ
  - ฟังก์ชันนี้จะแสดงชุดของเส้นเชื่อมที่มีค่าเท่ากับ 1 ในอาร์เรย์ "graph"
  - มีการใช้ลูปซ้อนลูปเพื่อวนลูปผ่านทุกคู่โหนดเพื่อตรวจสอบว่ามีเส้นเชื่อมระหว่างโหนดคู่นั้นหรือไม่ และถ้ามี เราจะแสดงคูโหนดนั้นในรูปแบบ (โหนดต้น, โหนดปลาย)
  - ข้อมูลเส้นเชื่อมทั้งหมดจะแสดงอยู่ในเครื่องหมาย "{}"

```

int main()
{
    printf("GRAPH (ADJACENCY MATRIX REPRESENTATION METHOD)\n");
    printf("=====\n");
    DispArray2D();
    DispSetOfVertex();
    DispSetOfEdge();
    getch();
    return(0);
} //End Main

```

- แสดงข้อความ "GRAPH (ADJACENCY MATRIX REPRESENTATION METHOD)" บนหน้าจอ
- เรียกใช้ฟังก์ชัน "DispArray2D" เพื่อแสดงค่าในอาร์เรย์สองมิติ "graph" และชื่อโหนด.
- เรียกใช้ฟังก์ชัน "DispSetOfVertex" เพื่อแสดงชุดของโหนดในกราฟ.
- เรียกใช้ฟังก์ชัน "DispSetOfEdge" เพื่อแสดงชุดของเส้นเชื่อมในกราฟ.
- getch() รอรับอักขระจากผู้ใช้ เพื่อปิดหน้าจอหรือออกจากโปรแกรม.
- สิ้นสุดการทำงานของฟังก์ชัน main() และโปรแกรมจะสิ้นสุดการทำงานโดยคืนค่า 0 เป็นสัญญาณว่าโปรแกรมทำงานเสร็จสมบูรณ์และไม่มีข้อผิดพลาด.

## ผลลัพธ์การทดลอง

```
C:\Users\Sarin\Desktop\ENG C  ×  +  ∨  
GRAPH (ADJACENCY MATRIX REPRESENTATION METHOD)  
=====
```

	A	B	C	D	E	F
A	0	1	1	1	0	0
B	1	0	1	0	1	0
C	1	1	0	0	0	0
D	1	0	0	0	1	1
E	0	1	0	1	0	0
F	0	0	0	1	0	0

```
Set of Vertex = {A,B,C,D,E,F}  
Set of Edge = {(A,B),(A,C),(A,D),(B,A),(B,C),(B,E),(C,A),(C,B),(D,A),(D,E),(D,F),(E,B),(E,D),(F,D),}
```

## สรุปผลการทดลอง

โปรแกรมนี้เป็นโปรแกรมที่ใช้ในการแสดงข้อมูลเกี่ยวกับกราฟที่มีการแทนด้วยเมทริกซ์เชื่อมโหนด (adjacency matrix)

## สื่อ / เอกสารอ้างอิง

ไฟล์ประกอบการสอนสัปดาห์ที่ 8 ของ อาจารย์ ปิยพล ยืนยงสถาวร เรื่อง : โครงสร้างข้อมูลกราฟ