Biểu diễn bằng ma trận kề (ma trận trọng số)

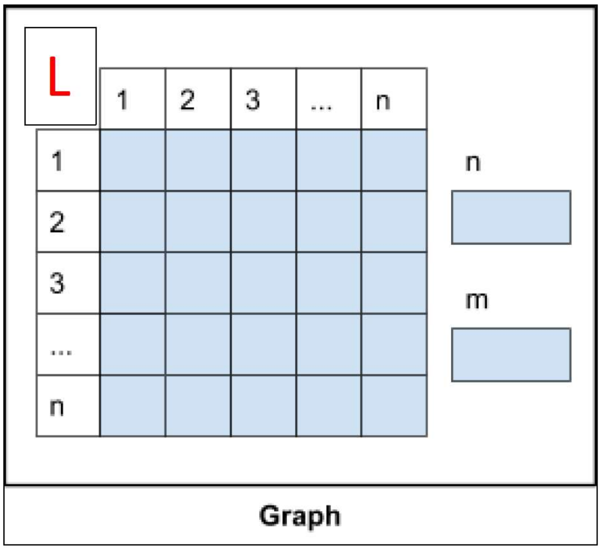
- **Ý tưởng:**

    - Lưu lại sự kề nhau của các cặp đỉnh (quan hệ giữa đỉnh và đỉnh)
    - Dùng một ma trận để lưu sự kề nhau của các cặp đỉnh (u, v) với giá trị 0 (NO_EDGE, nếu cung không tồn tại), với giá trị bằng trọng số cung (nếu cung tồn tại)
    - Nếu **đỉnh v là đỉnh kề của đỉnh u** thì phần tử ở **hàng u, cột v sẽ có giá trị trọng số nằm trên cung (u, v)** ngược lại có giá trị 0.

- **Chú ý:**

    - Đồ thị vô hướng sẽ có ma trận kề đối xứng
    - Không lưu trữ được đa cung (phải dùng phương pháp ma trận kề mở rộng)

- **Sơ đồ tổ chức dữ liệu (dùng mảng 2 chiều để lưu ma trận, bỏ qua cột 0 và hàng 0):**



Giải thuật Moore-Dijkstra:

Giải thuật Moore – Dijkstra cho phép tìm đường đi ngắn nhất từ s đến các đỉnh còn lại trên một đồ thị vô hướng (hoặc có hướng) có trọng số.

Ý tưởng:

- Khởi tạo đường đi ngắn nhất trực tiếp từ s đến các đỉnh còn lại.
- Sau đó lần lượt cập nhật lại đường đi nếu đường đi mới tốt hơn đường đi cũ.

Các biến hỗ trợ:

- pi[u]: chiều dài đường đi ngắn nhất từ s đến u (tính đến thời điểm đang xét).
- p[u]: đỉnh trước đỉnh u trên đường đi ngắn nhất từ s đến u (được sử dụng để dựng lại cây đường đi)
- L[u][v] bằng chiều dài (trọng số) của cung (u, v) nếu cung (u, v) tồn tại và L[u][v] = NO_EDGE nếu cung (u, v) KHÔNG tồn tại.
- mark[u]: cho biết đỉnh u đã được đánh dấu chưa. mark[u]=1 đã được đánh dấu, mark[u]=0 chưa được đánh dấu.

Giải thuật:

Khởi tạo:

- pi[u] = ∞ với mọi u;
- pi[s] = 0; (s là đỉnh bắt đầu)
- p[s]=-1;
- mark[u] = 0 với mọi u

Lặp (n – 1 lần)

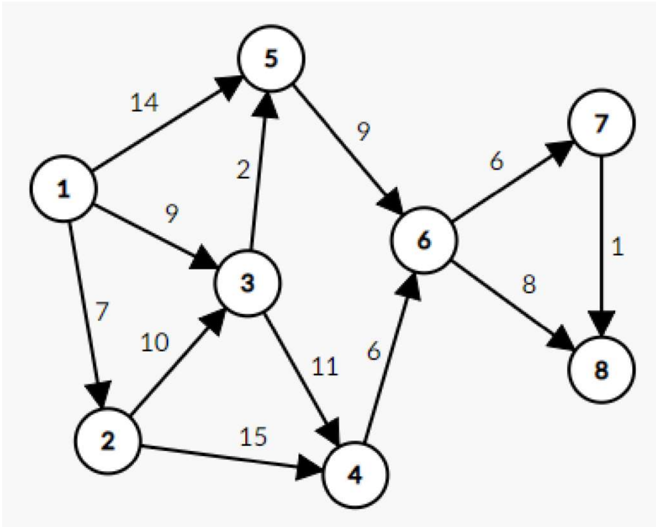
- Chọn đỉnh u chưa đánh dấu (mark[u] == 0) có chiều dài của đường đi từ s đến nó (pi[u]) nhỏ nhất.
- Đánh đấu đã xét u bằng cách đặt mark[u] = 1
- Xem xét cập nhật pi[v] và p[v] các đỉnh kề của u chưa được đánh dấu (mark[v] == 0)
    - Đỉnh v sẽ được cập nhật nếu đường đi mới (thông qua u) tốt hơn đường đi cũ:

```
if (pi[u] + L[u][v] < pi[v]) {
        pi[v] = pi[u] + L[u][v]    //cập nhật chiều dài đường đi ngắn nhất
        p[v] = u                   //cập nhật đường đi ngắn nhất
}
```

Cài đặt giải thuật:

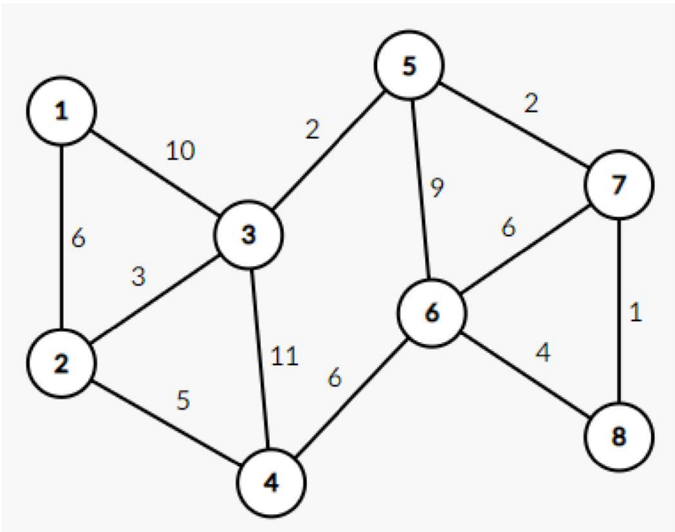
```
#define INFINITY 9999999

int mark[MAXN];
int pi[MAXN];
int p[MAXN];

void Dijkstra(Graph *G, int s) {
  int u, v, it;
  for (u = 1; u <= G->n; u++) {
    pi[u] = INFINITY;
    mark[u] = 0;
  }

  pi[s] = 0;
  p[s] = -1; //trước đỉnh s không có đỉnh nào cả

  //lặp n hoặc n-1 lần đều được
  for (it = 1; it < G->n; it++) {
    //1. Tìm j có mark[j] == 0 và có pi[j] nhỏ nhất và gán u=j
    int j, min_pi = INFINITY;
    for (j = 1; j <= G->n; j++)
      if (mark[j] == 0 && pi[j] < min_pi) {
        min_pi = pi[j];
        u = j;
      }
    //2. Đánh dấu u đã xét
    mark[u] = 1;

    //3. Cập nhật pi và p của các đỉnh kề của u (nếu thoả)
    for (v = 1; v <= G->n; v++)
      if (G->L[u][v] != NO_EDGE && mark[v] == 0)
        if (pi[u] + G->L[u][v] < pi[v]) {
          pi[v] = pi[u] + G->L[u][v];
          p[v] = u;
        }
  }
}
```

Câu 1: Cho đồ thị có hướng như hình sau:



a. Hãy vẽ sơ đồ tổ chức dữ liệu cho đồ thị trên bằng cách sử dụng phương pháp ma trận trọng số.
b. Hãy chạy thủ công giải thuật Moore-Dijkstra để tìm đường đi ngắn nhất từ đỉnh 1 đến tất cả các đỉnh còn lại.
c. Dựa vào giá trị của p[i]. Hãy dựng lại cây đường đi.
d. Hãy cho biết đường đi ngắn nhất từ đỉnh 1 đến đỉnh 8 là bao nhiêu?
e. Dựa vào cây đường đi. Hãy cho biết, để đi từ đỉnh 1 đến đỉnh 7, ta cần phải qua những đỉnh nào để tổng độ dài đường đi là ngắn nhất.

Câu 2: Cho đồ thị vô hướng như hình sau:



a. Hãy vẽ sơ đồ tổ chức dữ liệu cho đồ thị trên bằng cách sử dụng phương pháp ma trận trọng số.
b. Hãy chạy thủ công giải thuật Moore-Dijkstra để tìm đường đi ngắn nhất từ đỉnh 1 đến tất cả các đỉnh còn lại.
c. Dựa vào giá trị của p[i]. Hãy dựng lại cây đường đi.
d. Hãy cho biết đường đi ngắn nhất từ đỉnh 1 đến đỉnh 5 là bao nhiêu?
e. Hãy cho biết đường đi ngắn nhất từ đỉnh 1 đến đỉnh 6 là bao nhiêu?
f. Hãy cho biết đường đi ngắn nhất từ đỉnh 1 đến đỉnh 7 là bao nhiêu?
g. Hãy cho biết đường đi ngắn nhất từ đỉnh 1 đến đỉnh 8 là bao nhiêu?
h. Dựa vào cây đường đi. Hãy cho biết, để đi từ đỉnh 1 đến đỉnh 7, ta cần phải qua những đỉnh nào để tổng độ dài đường đi là ngắn nhất.