**TRƯỜNG CAO ĐẲNG CÔNG NGHỆ THỦ ĐỨC**

**KHOA CÔNG NGHỆ THÔNG TIN**



**BÁO CÁO ĐỒ ÁN:**

* **TÌM ĐƯỜNG ĐI NGẮN NHẤT– THUẬT TOÁN A\***
* **TÌM TÔ MÀU – THUẬT TOÁN GRAPH COLORING**

**SINH VIÊN: NGUYỄN LÊ TRỌNG TIỀN**

**MSSV: 19211TT4165**

**Mail: baobaote00@gmail.com**

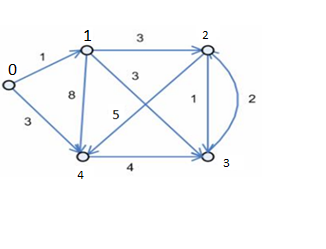
31/12/2020

**TÌM CÂY PHỦ TỐI THIỂU – THUẬT TOÁN KRUSKAL**

-Ý tưởng:

1. Tìm đường đi ngắng nhất qua tất cả các cặp đỉnh bằng thuật toán floyd
2. Cho Open = PriorityQueue, Closed = PriorityQueue.
3. Đặt giá trị heurictic của đỉnh đầu.
4. Thêm đỉnh đầu vào Open.
5. Nếu Open rỗng thì trả về không tìm được.
6. Cho O = Open.Dequeue() và thêm O vào Closed.
7. Nếu O bằng với đỉnh cuối thì trả về đường đi.
8. Tạo node cho các đỉnh kề với O là tmp và gắn đỉnh trước bằng đỉnh O nếu tmp không nằm trong Open hoặc Closed.
9. Lặp lại bước 6.

-Ví dụ:



Đỉnh bắt đầu: 0 -> 2

Bước chuẩn bị:

Danh sách kề Heuristic[,]:

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 0 | 1 | 1 | 4 | 3 | 0 |  |  |
| 1 | 2 | 3 | 3 | 3 | 4 | 8 | 1 |
| 2 | 3 | 1 | 4 | 5 | 4 |  |  |
| 3 | 2 | 2 | 4 |  |  |  |  |
| 4 | 3 | 4 | 3 |  |  |  |  |

\*chú thích

|  |  |  |
| --- | --- | --- |
|  |  |  |
| Cạnh kề | Trọng số | Heuristic |

\*node: 0,0,0 đỉnh,trọng số,heuristic

Danh sách Open:

|  |
| --- |
| 0,0,0 |

Danh sách Closed:

|  |
| --- |
| null |

1. Lần lặp 1:

* Bước 1: duyệt node 0,0,0
* Bước 2: thêm node 0,0,0 vào Closed

Danh sách Closed:

|  |
| --- |
| 0,0,0 |

* Bước 3: tạo các node cho đỉnh kề[1,4], gán node trước bằng 0,0,0 và thêm đỉnh[1,4] vào Open

Danh sách Open:

|  |  |
| --- | --- |
| 1,1,1 | 4,3,3 |

1. Lần lặp 2:

* Bước 1: duyệt node 1,1,1
* Bước 2: thêm node 1,1,1 vào Closed

Danh sách Closed:

|  |  |
| --- | --- |
| 0,0,0 | 1,1,1 |

* Bước 3: tạo các node cho đỉnh kề[2,3,4] ], gán node trước bằng 1,1,1 và thêm đỉnh[2,3] vào Open

Danh sách Open:

|  |  |  |
| --- | --- | --- |
| 4,3,3 | 2,4,4 | 3,4,4 |

1. Lần lặp 3:

* Bước 1: duyệt node 4,3,3
* Bước 2: thêm node 4,3,3 vào Closed

Danh sách Closed:

|  |  |  |
| --- | --- | --- |
| 0,0,0 | 1,1,1 | 4,3,3 |

* Bước 3: tạo các node cho đỉnh kề[3] và thêm đỉnh[] vào Open

Danh sách Open:

|  |  |
| --- | --- |
| 3,4,4 | 2,4,4 |

1. Lần lặp 4:

* Bước 1: duyệt node 3,4,4
* Bước 2: thêm node 3,4,4 vào Closed

Danh sách Closed:

|  |  |  |  |
| --- | --- | --- | --- |
| 0,0,0 | 1,1,1 | 4,3,3 | 3,4,4 |

* Bước 3: tạo các node cho đỉnh kề[2] và thêm đỉnh[] vào Open

Danh sách Open:

|  |
| --- |
| 2,4,4 |

1. Lần lặp 5:

* Bước 1: duyệt node 2,4,4
* Bước 2: node 2,4,4 bằng với điểm cuối trả về đường đi

Đường đi 0->1->2 , tổng chi phí 4

**TÔ MÀU ĐỒ THỊ**

**THUẬT TOÁN WELCH-POWELL**

-Ý tưởng:

+ Sắp xếp các đỉnh G theo bậc giảm dần.

+ Dùng một màu để tô đỉnh đầu tiên và cũng dùng màu này để tô màu các đỉnh liên tiếp trong danh sách mà không kề với đỉnh đầu tiên.

+ Bắt đầu trở lại đầu danh sách, tô màu thứ hai cho đỉnh chưa được tô và lập lại quá trình trên cho đến khi tất cả các đỉnh đều được tô màu.

-Ví dụ:



Bước chuẩn bị:

Danh sách đỉnh:

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | V1 | V2 | V3 | V4 | V5 | V6 |
| Màu | -1 | -1 | -1 | -1 | -1 | -1 |
| Bậc | 4 | 2 | 4 | 3 | 2 | 3 |

Danh sách đỉnh đã được sắp xếp:

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | V1 | V3 | V4 | V6 | V2 | V5 |
| Màu | -1 | -1 | -1 | -1 | -1 | -1 |
| Bậc | 4 | 4 | 3 | 3 | 2 | 2 |

1. Lần lặp 1:

* Tô màu 0 cho đỉnh v1 và v5

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | V1 | V3 | V4 | V6 | V2 | V5 |
| Màu | 0 | -1 | -1 | -1 | -1 | 0 |
| Bậc | 4 | 4 | 3 | 3 | 2 | 2 |

1. Lần lặp 2:

* Tô màu 1 cho đỉnh v3 và v6

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | V1 | V3 | V4 | V6 | V2 | V5 |
| Màu | 0 | 1 | -1 | 1 | -1 | 0 |
| Bậc | 4 | 4 | 3 | 3 | 2 | 2 |

1. Lần lặp 3:

* Tô màu 2 cho đỉnh v4 và v2

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | V1 | V3 | V4 | V6 | V2 | V5 |
| Màu | 0 | 1 | -1 | 1 | -1 | 0 |
| Bậc | 4 | 4 | 3 | 3 | 2 | 2 |