**Bài toán tô màu đồ thị bằng phương pháp tham lam**

1. Nêu bài toán

Tô mỗi đỉnh của đồ thị bằng một màu sao cho hai đỉnh kề nhau có màu khác nhau. Tìm cách tô tất cả các đỉnh cả đồ thị với số màu ít nhất

1. Ý tưởng bài toán

Ý tưởng – Qui ước màu là các số: 1, 2, 3, ...

1. Tô màu một đỉnh bất kì với màu 1
2. Với đỉnh v chưa tô màu: Tô nó với màu là số nhỏ nhất chưa dùng với các đỉnh kề với nó(các đỉnh này đã được tô màu), đỉnh v đã được tô màu.
3. Lặp lại b cho đến khi tất cả các đỉnh đã được tô màu.

Giả mã:

//giả sử ta chọn số đỉnh chính là số thứ tự của mỗi đỉnh

// chọn đỉnh đầu tiên và tô đỉnh đầu tiên với màu thứ nhất

//mauphu[] là mảng lưu các màu đã được tô cho các đỉnh kề với đỉnh đang xét

t = 1, j = 2; // t là xét số đỉnh đã được tô màu và kề với đỉnh đang xét, j là đỉnh đang xet

Mau = 1;

Maudinh[1] = mau; //gán màu cho đỉnh

Dinhtruoc[1] = 1;// đỉnh đã tô đưa vào mảng đỉnh trước

Do {

For (int k=1; k<j;i++)

{

If{a[j][dinhtruoc[k]] == 1}

Mauphu[++t]=maudinh[dinhtruoc[k]];

}

/\* gọi hàm chonmau để tô màu cho đỉnh j;hàm chonmau làm nhiệm vụ chọn màu phù hợp khác màu các đỉnh đã tô mà kề với đỉnh đang xet j.\*/

Maudinh[j]=chonmau(mauphu,t)

//đỉnh vừa tô được đưa vào mảng đỉnh trước

Dinhtruoc[l]=j;

//cập nhật lại các giá trị khác

j++,t=1;

}while (j<=n); // còn đỉnh chưa tô

1. Bộ dữ liệu

Tất cả các đồ thị đều được lưu dưới file txt bằng ma trận kề

1. Bộ thứ nhât

9

6

3

1

8

4

10

7

5

2

Ma trận kề:

0 1 1 0 0 0 0 0 0 0

1 0 1 0 1 0 0 0 0 0

1 1 0 1 1 1 0 0 0 0

0 0 1 0 1 1 0 0 0 0

0 1 1 1 0 0 1 0 0 0

0 0 1 1 0 0 1 1 1 0

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Đỉnh | kề với đỉnh đang xét(đã tô màu) | Số màu đã chọn | Màu chọn cho đỉnh | Đỉnh trước (dinhtruoc[]) |
| 1 |  | 1 | 1 | 1 |
| 2 | 1 | 1,2 | 2 | 1,2 |
| 3 | 1,2 | 1,2,3 | 3 | 1,2,3 |
| 4 | 3 | 1,2,3 | 1 | 1,2,3,4 |
| 5 | 2,3,4 | 1,2,3,4 | 4 | 1,2,3,4,5 |
| 6 | 3,4 | 1,2,3,4 | 2 | 1,2,3,4,5,6 |
| 7 | 5,6 | 1,2,3,4 | 1 | 1,2,3,4,5,6,7 |
| 8 | 6 | 1,2,3,4 | 1 | 1,2,3,4,5,6,7,8 |
| 9 | 6,8 | 1,2,3,4 | 3 | 1,2,3,4,5,6,7,8,9 |
| 10 | 7,9 | 1,2,3,4 | 2 | 1,2,3,4,5,6,7,8,9,10 |

Chọn đỉnh 1 là đỉnh xét đầu tiên, tô đỉnh 1 với màu 1, đỉnh 1 đã xét, số màu được chọn hiện tại là 1

Xét đỉnh 2: kiểm tra đỉnh 2 với các tập đã xét, ta thấy đỉnh 2 kề với đỉnh 1 đã được xét, đỉnh 1 đã được tô màu số 1, vậy thêm màu mới để tô màu cho đỉnh 2, vậy số màu được chọn hiện tại là 2

Xét đỉnh 3: kiểm tra đỉnh 3 với các đỉnh đã xét, đỉnh 3 kề với 1 và 2, tăng số màu lên 3 tô cho đỉnh 3, số màu hiện tại là 3

Xét đỉnh 4: kiểm tra thấy rằng đỉnh 4 kề với đỉnh 3 đã xét, đỉnh 3 được tô màu số 3, vậy đỉnh 4 có thể tô màu số 1

Xết đỉnh 5: đỉnh 5 kề với 2,3,4; đỉnh 2 tô màu 2, 3 tô màu 3, 4 tô màu 1, vậy cần thêm 1 màu mới tô đỉnh 5, số màu cần dùng hiện tại là 4

Xét đỉnh 6: đỉnh 6 kề với 3,4 ; đỉnh 3 được tô màu 3, 4 tô màu 1, vậy đỉnh 6 được tô bằng màu số 2

Xét đỉnh 7: đỉnh 7 kề với 5,6; đỉnh 5 được tô màu 5, 6 tô màu 2, vậy đỉnh 7 được tô với màu số 1

Xét đỉnh 8: đỉnh 8 kề với 6; đỉnh 6 được to màu 2, vậy đỉnh 8 được tô với màu số 1

Xét đỉnh 9: kề với 9 đã được tô màu là 6,8; đỉnh 6 được tô bằng màu 2; đỉnh 8 tô màu 1=> đỉnh 9 được tô màu số 3

Xét đỉnh 10: kề với 10 là 7, 9; đỉnh 7 tô màu 1, đỉnh 9 tô màu 3, vậy đỉnh 10 được tô màu 2

* Quá trình chọn màu: xét màu của các đỉnh kề với số thứ tự màu đã được tô từ số màu bé nhất, nếu màu có số thứ tự đó chưa được tô cho các đỉnh kề đã xét của đỉnh đang xét thì dùng màu đó tô cho đỉnh đang xét.

1. Bộ dữ liệu 2

Làm tương tự như phần a

6

2

5

8

4

1

10

3

9

7

|  |  |
| --- | --- |
| 10  0 1 1 0 0 0 0 0 0 0  1 0 1 1 0 0 0 0 0 0  1 1 0 1 0 0 0 0 0 0  0 1 1 0 1 0 1 0 0 0  0 0 0 1 0 1 1 0 0 0  0 0 0 0 1 0 0 1 1 1  0 0 0 1 1 0 0 1 1 0  0 0 0 0 0 1 1 0 1 0  0 0 0 0 0 1 1 1 0 1  0 0 0 0 0 1 0 0 1 0 | mau dinh 1: 1  mau dinh 2: 2  mau dinh 3: 3  mau dinh 4: 1  mau dinh 5: 2  mau dinh 6: 1  mau dinh 7: 3  mau dinh 8: 2  mau dinh 9: 4  mau dinh 10: 2 |

1. Bộ thứ 3

9

5

4

1

111

12

6

3

10

8

7

2

|  |  |
| --- | --- |
| 12  0 1 1 1 0 0 0 0 0 0 0 0  1 0 1 0 0 1 1 0 0 0 0 0  1 1 0 0 0 1 0 0 0 0 0 0  1 0 0 0 1 0 0 0 0 0 0 0  0 0 0 1 0 0 0 0 1 1 0 1  0 1 1 0 0 0 0 1 0 0 0 0  0 1 0 0 0 0 0 1 0 0 0 0  0 0 0 0 0 1 1 0 0 1 0 0  0 0 0 0 1 0 0 0 0 0 1 1  0 0 0 0 1 0 0 1 0 0 1 0  0 0 0 0 0 0 0 0 1 1 0 0  0 0 0 0 1 0 0 0 1 0 0 0 | mau dinh 1: 1  mau dinh 2: 2  mau dinh 3: 3  mau dinh 4: 2  mau dinh 5: 1  mau dinh 6: 1  mau dinh 7: 1  mau dinh 8: 2  mau dinh 9: 2  mau dinh 10: 3  mau dinh 11: 1  mau dinh 12: 3 |

1. Bộ thứ 4

111

9

5

4

1

6

8

3

7

10

2

|  |  |
| --- | --- |
| 11  0 1 1 1 0 0 0 0 0 0 0  1 0 1 0 0 0 1 0 0 0 0  1 1 0 0 1 0 0 0 0 0 0  1 0 0 0 1 0 0 0 0 0 0  0 0 0 1 0 1 0 1 1 0 0  0 0 0 0 1 0 1 0 0 0 0  0 1 0 0 0 1 0 0 0 1 0  0 0 0 0 1 0 0 0 0 1 0  0 0 0 0 1 0 0 0 0 1 0  0 0 0 0 0 0 1 1 0 0 1  0 0 0 0 0 0 0 0 1 1 0 | mau dinh 1: 1  mau dinh 2: 2  mau dinh 3: 3  mau dinh 4: 2  mau dinh 5: 1  mau dinh 6: 2  mau dinh 7: 1  mau dinh 8: 2  mau dinh 9: 2  mau dinh 10: 3  mau dinh 11: 1 |

1. Bộ thứ 5

9

6

3

5

1

7

4

8

2

10

|  |  |
| --- | --- |
| 10  0 1 1 1 0 0 0 0 0 0  1 0 0 1 0 0 0 1 0 0  1 0 0 0 1 0 0 0 0 0  1 1 0 0 1 0 0 1 0 0  0 0 1 1 0 1 1 0 0 0  0 0 0 0 1 0 0 0 1 0  0 0 0 0 1 0 0 0 0 1  0 1 0 1 0 0 0 0 0 1  0 0 0 0 0 1 0 0 0 1  0 0 0 0 0 0 1 1 1 0 | mau dinh 1: 1  mau dinh 2: 2  mau dinh 3: 2  mau dinh 4: 3  mau dinh 5: 1  mau dinh 6: 2  mau dinh 7: 2  mau dinh 8: 1  mau dinh 9: 1  mau dinh 10: 3 |

1. Độ phức tạp của thuật toán:

Độ phức tạp thuật toán của mã nguồn trên là O(n^2), với n là số đỉnh của đồ thị.

Trong hàm tomau(), vòng lặp bên ngoài chạy từ đỉnh thứ hai đến đỉnh thứ n, nên sẽ có n-1 lượt lặp. Trong mỗi lượt lặp này, lại có một vòng lặp trong đó ta duyệt qua tất cả các đỉnh kề của đỉnh đang xét, đồng thời kiểm tra màu của các đỉnh kề đã được sử dụng hay chưa (bằng cách gọi hàm kiemtramau()). Việc kiểm tra này cũng đòi hỏi phải duyệt qua tất cả các đỉnh kề, do đó độ phức tạp của vòng lặp này là O(n).

Tổng cộng, độ phức tạp của hàm tomau() là O(n^2), do đó độ phức tạp của toàn bộ chương trình sẽ là O(n^2) (do hàm tomau() được gọi trong hàm main()).