

Раскраска графа

Шамакова Екатерина

shamaich0168@gmail.com

План:

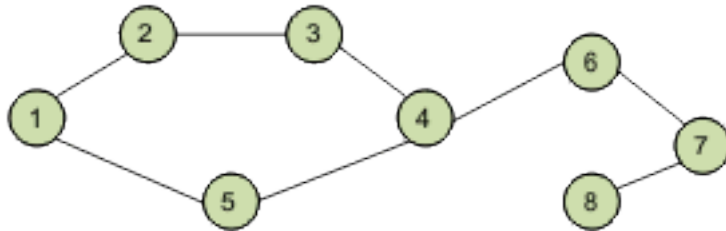
- ▶ Что такое граф и основные определения, связанные с ним
- ▶ Постановка задачи
- ▶ Алгоритмы реализации раскраски графа
 - ▶ Полный перебор
 - ▶ Жадный алгоритм
- ▶ Инструкция пользователя
- ▶ Выводы

Основные определения

Графом называют математическую модель, представляющую собой множество вершин и набор рёбер

Вершины: 1, 2, 3, 4, 5, 6, 7, 8

Рёбра: 1-2, 1-5, 2-3, 3-4, 5-4 ...



Две вершины считаются **смежными**, если имеют хотя бы одно общее ребро

Основные определения

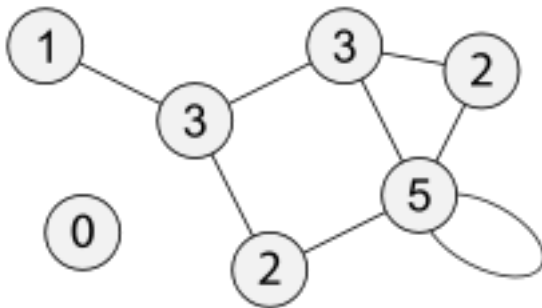


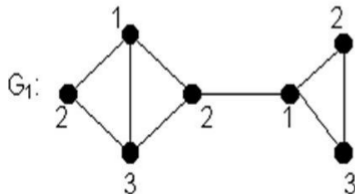
Рис. 1: Граф, на вершинах которого отмечены степени

Степень или **валентность** вершины графа — это число ребер, входящих в эту вершину.

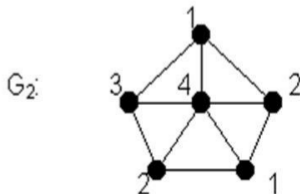
Основные определения

Наименьшее число красок, необходимое для правильной раскраски графа G называется **хроматическим числом** графа G . Хроматическое число обозначается через $\chi(G)$.

Пример



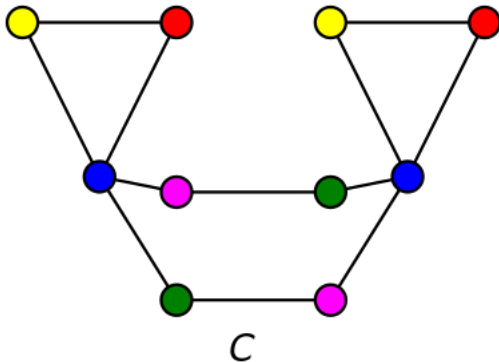
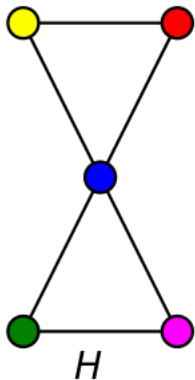
$$\chi(G_1) = 3$$



$$\chi(G_2) = 4$$

Раскраска графа

- ▶ Раскрасить вершины графа
- ▶ Любые 2 смежные вершины имеют разные цвета
- ▶ Используя минимальное количество цветов

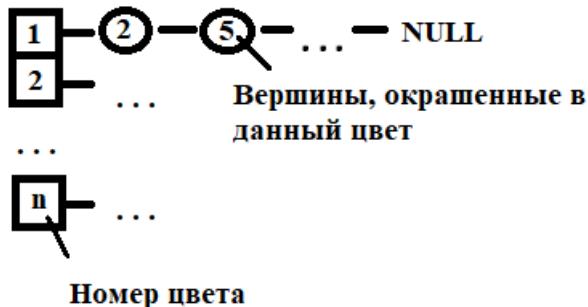


Алгоритмы раскраски графа

- ▶ Полный перебор
- ▶ Жадный алгоритм

Алгоритмы раскраски графа

Способ хранения раскрашенных вершин: `vector< list< int > >`
table



Полный Перебор

Попробуем раскрасить граф, последовательно увеличивая количество цветов, начиная с 1, пока не останется нераскрашенных вершин. То число цветов, которым получится раскрасить граф, и будет его хроматическим числом.

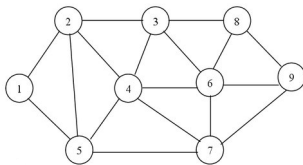


Рис. 2: Начальный граф

Этапы работы полного перебора

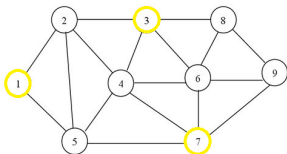


Рис. 3: Раскрашиваем его 1 цветом

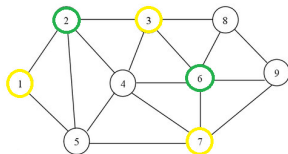


Рис. 4: Раскрашиваем его 2 цветами

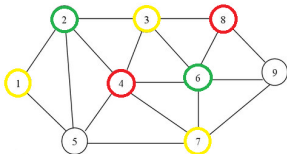


Рис. 5: Раскрашиваем его 3 цветами

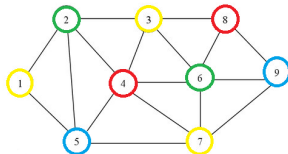


Рис. 6: Раскрашенный граф

Полный перебор (псевдокод)

Вызов из main

```
int main()
{
    ...
    bool done=false;    //проверка, все ли вершины смогн
                        и окрасить
    int max_color=1;
    while(done!=true){ //пока остались нераскрашенные
                        вершины
        done=brute_force(0, max_color);
        max_color++;    //увеличиваем число красок
        ...
    }
```

Полный перебор (псевдокод)

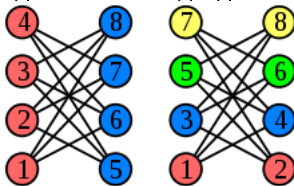
```
bool brute_force(int k, int max_color) {
    if(все вершины окрашены)
    {
        chromatic_number = max_color;
        return true;
    }
    else {
        for (от минимального цвета до максимального) {
            for (по списку вершин с одним цветом)
                //если не смежна ни с одной из них
                //раскрашиваем вершину

                brute_force(k + 1, max_color);
            if (окрасили последнюю вершину)
                //переходим к концу функции
            else
                //удаляем цвет
        }
    }
}
```

Жадный алгоритм

1. Сортируем вершины по их степеням в порядке убывания.
2. Последовательно окрашиваем вершины в выбранный цвет. Если у вершины уже есть смежные вершины с выбранным цветом, то оставляем ее неокрашенной.
3. Если остались неокрашенные вершины, то выбираем следующий цвет и возвращаемся ко 2 пункту.

Качество полученной раскраски зависит от выбранного порядка. Жадный не всегда даёт оптимальное решение.



Например:

Этапы работы жадного алгоритма

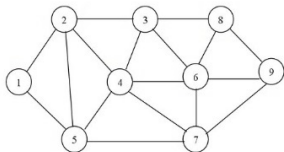


Рис. 7: Граф до работы алгоритма

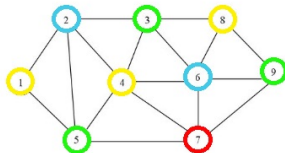


Рис. 8: Раскрашенный граф



Рис. 9: этапы работы жадного алгоритма

Жадный алгоритм (псевдокод)

```
vector< list <int> > greedy() {  
    //сортируем вершины графа в порядке убывания их сте  
    пеней  
    for (по всем вершинам) {  
        if (вершина не раскрашена) {  
            //добавляем её в список (присваиваем ей цвет)  
            for (int j = i + 1; j < size; j++) {  
                if (вершина не окрашена) {  
                    for (по списку вершин, принадлежащих од  
                        ному цвету) {  
                        //если не смежна ни с одной из вершин  
                        //раскрашиваем её  
                    }  
                }  
            }  
            //переходим к следующему цвету  
        }  
    }  
}
```

Инструкция пользователя

Необходимо ввести: in.txt out.gv algNum

- ▶ in.txt: текстовый файл, в котором первой строкой записано количество вершин графа, а далее – матрица смежности через пробел
- ▶ out.gv: файл, в который запишется раскрашенный граф в формате graphviz
- ▶ algNum: 0-полный перебор, 1-жадный алгоритм

Пример файла in.txt:

```
3
0  1  1
1  0  1
1  1  0
```

Далее в ходе выполнения программы необходимо будет ввести указанное количество цветов на английском языке через "Enter"

Вывод

Название алгоритма	Сложность	Качество
Жадный алгоритм	$O(n^3)$	есть контр. пример
Полный перебор	$O(n!)$	оптимальный