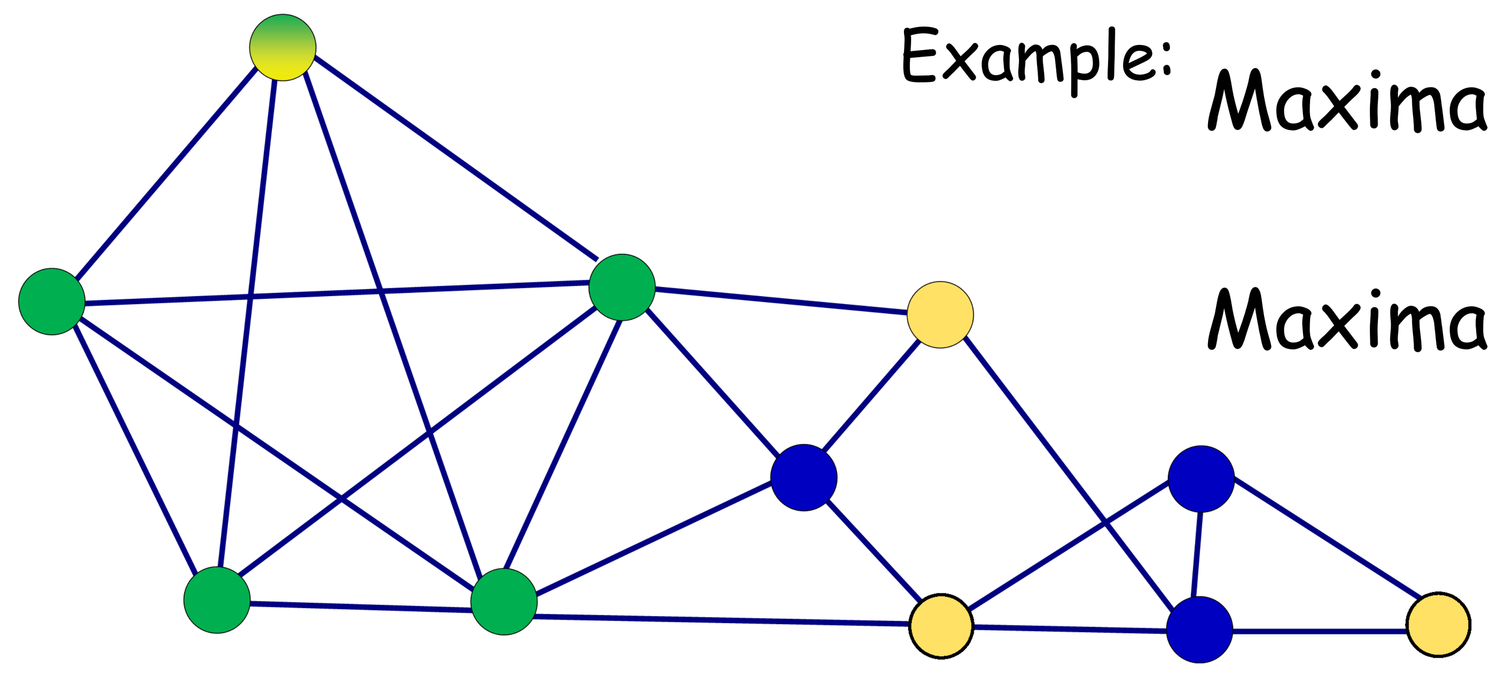
Algorithms for finding maximal cliques and their application in Sociology

Zubrilina Sofia

Lipetsk, Russia

Lyceum №44, Lipetsk Educational Center “Strategy”, ninth-grade student

An independent set is a set of pairwise nonadjacent vertices.
Clique in a graph is a set of pairwise adjacent vertices.



Example: Maximal clique is a set of green vertices.

Maximal independent set is a set of yellow vertices.

The problem of finding a maximal clique in a graph is NP-hard.

Ramsey's theorem

For any two natural numbers, s and t, there exists a natural number, R(s,t) = N, so any graph of size at least N must contain an independent set of size s or a clique of size t.

R(s, t) for different s and t.

r / s	1	2	3	4	5	6	7	8
1	**1**	1	1	1	1	1	1	1
2	1	**2**	3	4	5	6	7	8
3	1	3	**6**	9	14	18	23	28
4	1	4	9	**18**	25	36–41	49–61	58–84
5	1	5	14	25	**43–49**	58–87	80–143	101–216
6	1	6	18	36–41	58–87	**102–165**	113–298	132–495
7	1	7	23	49–61	80–143	113–298	**205–540**	217–1031
8	1	8	28	58–84	101–216	132–495	217–1031	**282–1870**

Algorithms for finding maximal cliques

Polynomial algorithms	Non polynomial algorithms
Greedy algorithm $O(n*\log n + n^2)$	Bron-Kerbosch algorithm $O(3^{n/3})$
Heuristic algorithm $O(n^2)$	Enumerative algorithm $O(2^n)$
Randomized algorithm $O(n^2)$	

I have implemented **Ramsey's algorithm**:

- recursion
- returns the maximal clique and the maximal independent set
- transforms graph into a binary tree where a root vertex is adjacent to all of its right descendants and non-adjacent to all of its left descendants

- complexity: $O(n^2)$
- development environment: Microsoft Visual Studio
- programming language: C#

$$I = \max(I_2 \cup \{v\}, I_1)$$
$$C = \max(C_1 \cup \{v\}, C_2)$$