Ю. В. Матиясевич

Я *Вам* докажу

https://logic.pdmi.ras.ru/~yumat

ТЕОРИЯ ДОКАЗАТЕЛЬСТВ
в гражданских и уголовных процессах

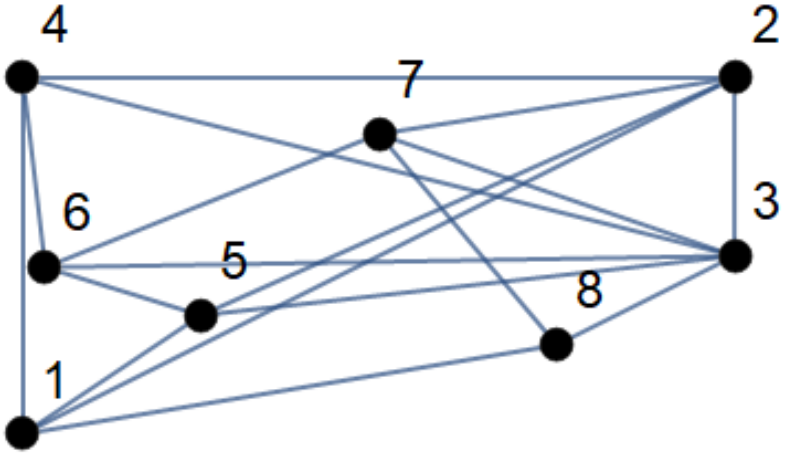
Свойства интерактивных доказательств

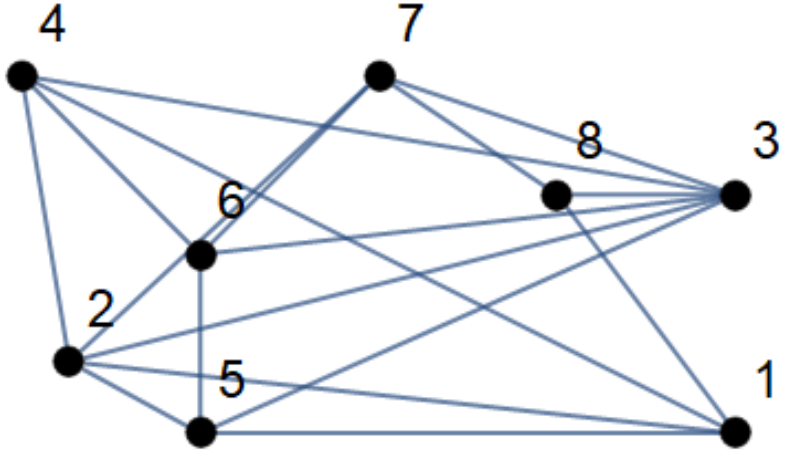
- Есть две стороны:

Prover	Verifier
Доказывающий	Проверяющий

- Доказательство обычно проходит в несколько раундов
- Проверяющий играет активную роль
- Проверяющий генерирует (обычно, случайным образом), некоторую информацию, недоступную Доказывающему
- Есть ничтожно маленькая вероятность обмана
- Посторонний наблюдатель не воспринимает интерактивное доказательство как убедительное
- Проверяющий, будучи убеждён лично, не в состоянии убедить кого-либо третьего

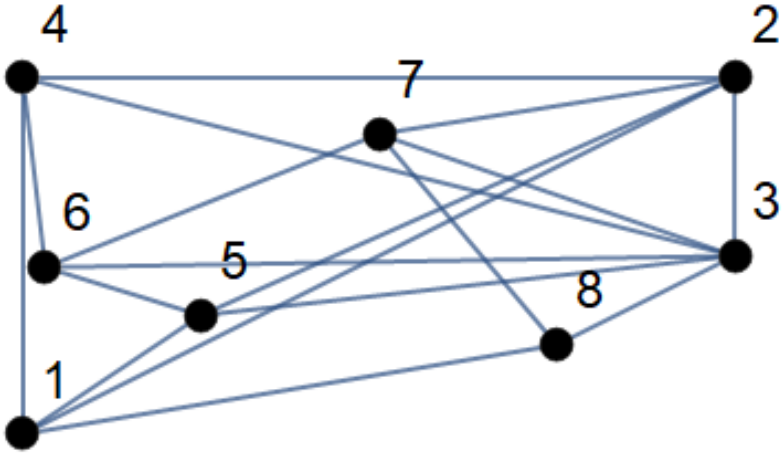
Изоморфизм графов

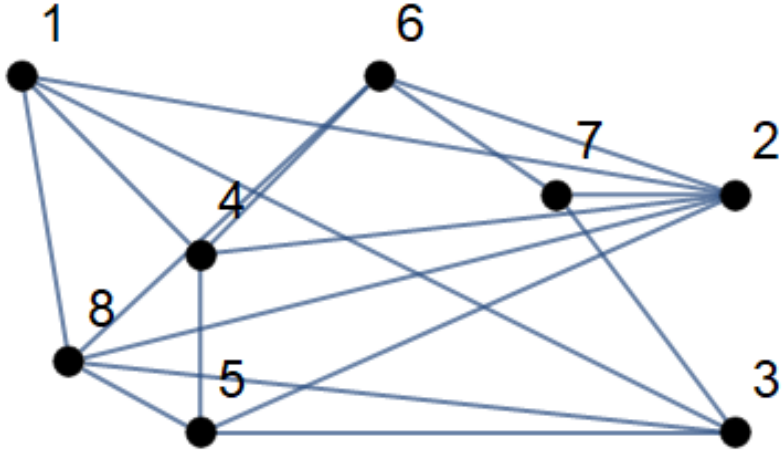




	1	2	3	4	5	6	7	8
1	0	0	0	0	1	1	0	1
2	0	0	0	1	1	0	0	0
3	0	0	0	0	0	0	1	1
4	0	1	0	0	0	0	1	1
5	1	1	0	0	0	1	1	1
6	1	0	0	0	1	0	1	0
7	0	0	1	1	1	1	0	0
8	1	0	1	1	1	0	0	0

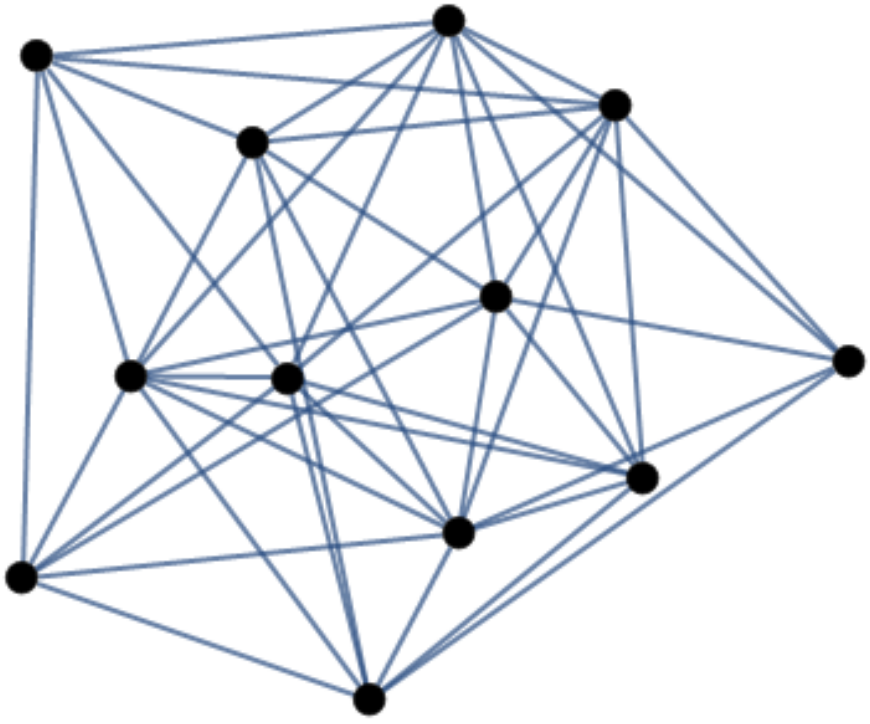
Изоморфизм графов

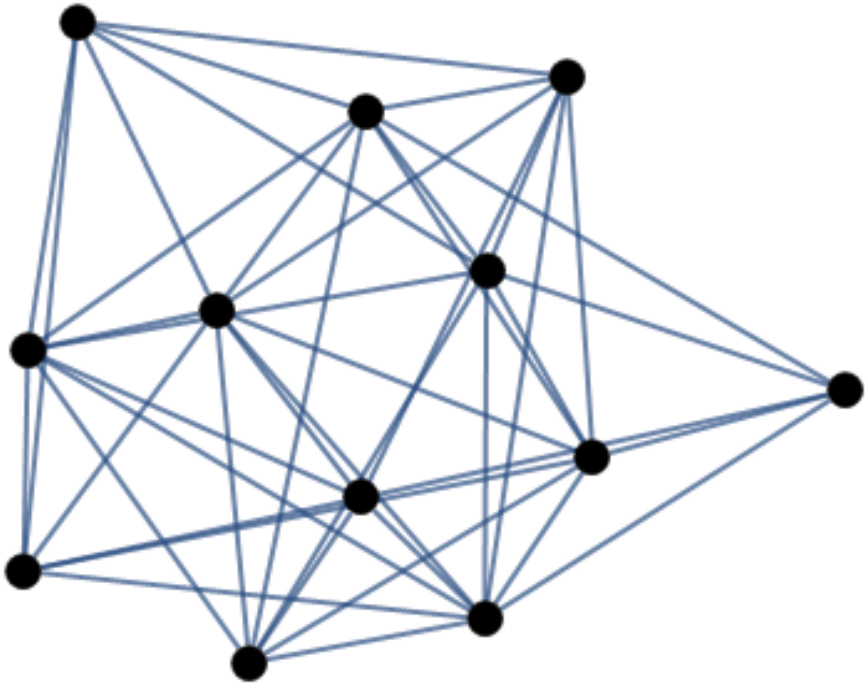




$$\begin{pmatrix} 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 \\ 5 & 2 & 4 & 6 & 3 & 7 & 1 & 8 \end{pmatrix}$$

Изоморфизм графов





Протокол доказательства неизомофности графов Γ_0 и Γ_1 с n вершинами

Каждый раунд состоит из 5 шагов:

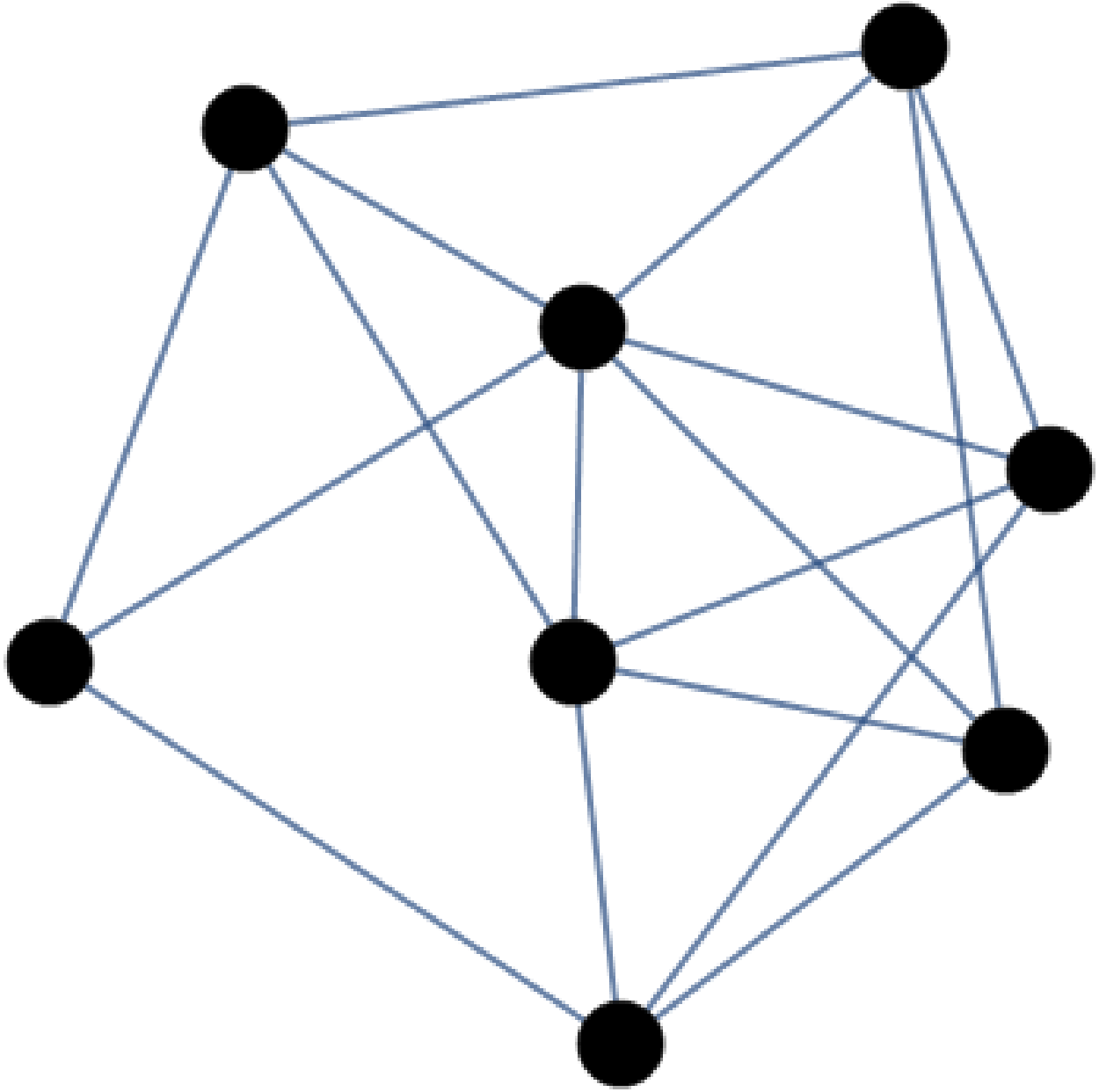
1. Проверяющий "кидает монетку" и определяет случайный бит m, $m = 0$ или $m = 1$
2. Проверяющий многократно "кидает монетку" и строит некоторую случайную подстановку
3. С помощью этой подстановки Проверяющий строит граф Γ как изоморфный образ графа Γ_m, и передаёт граф Γ Доказывающему
4. Доказывающий определяет, какому из двух графов, графу Γ_0 или графу Γ_1, изоморфен граф Γ, и сообщает это Проверяющему
5. Проверяющий определяет, правилен ли ответ Доказывающего

Требуемые свойства протокола

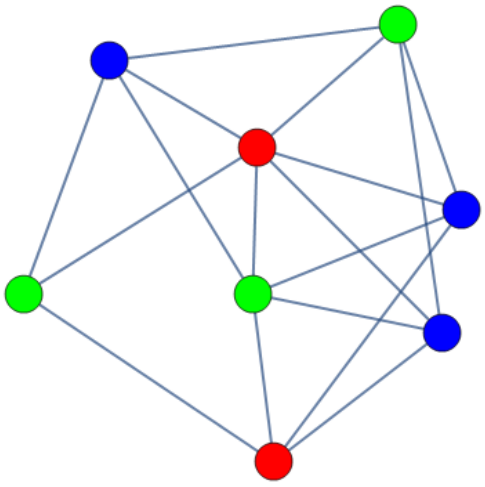
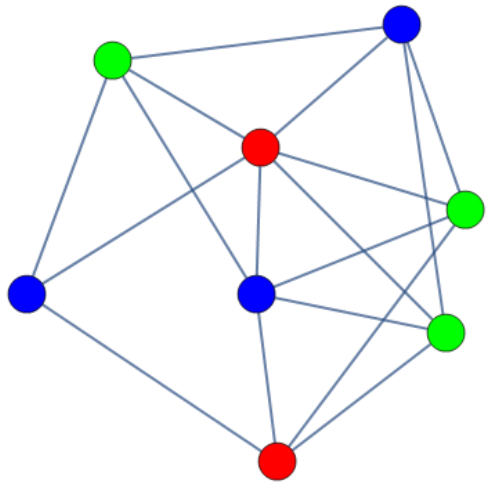
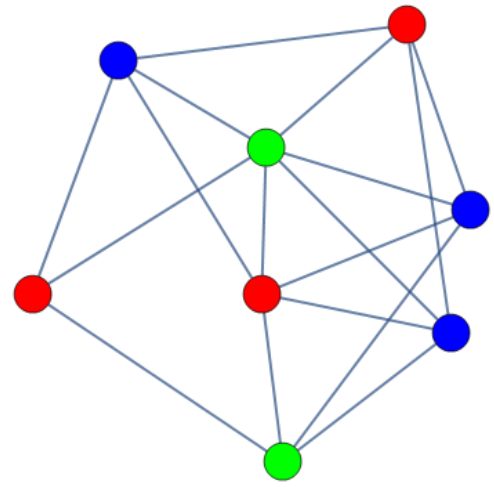
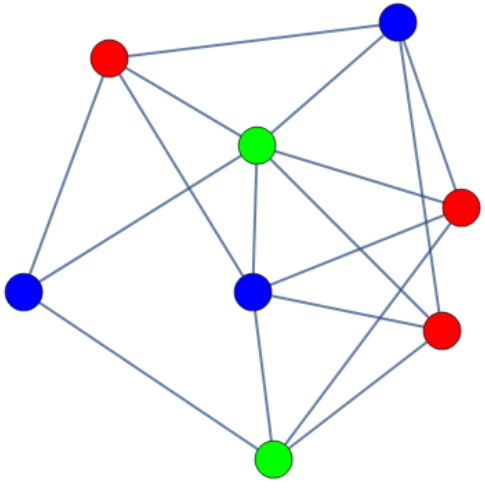
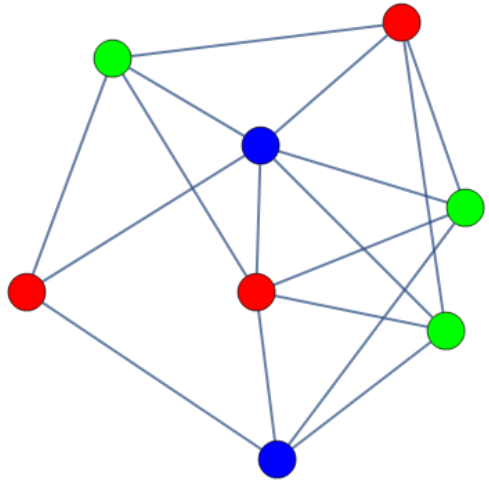
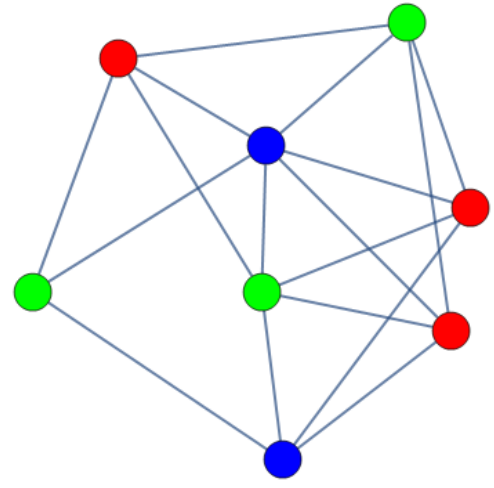
Полнота. Если доказываемое утверждение верно, то Доказывающий имеет возможность правильно ответить на все вопросы Проверяющего

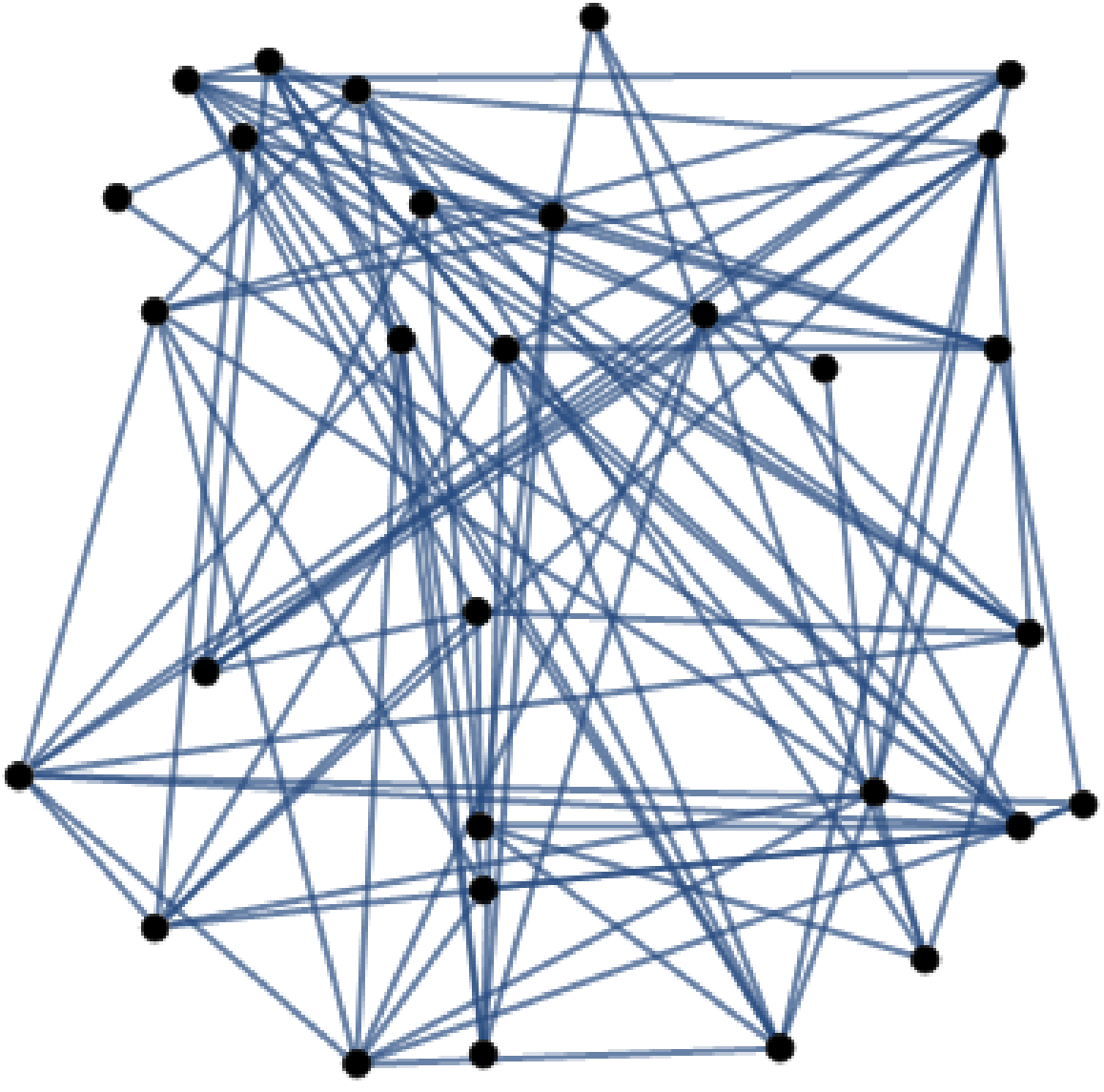
Корректность. Если доказываемое утверждение не является верным, то что бы (нечестный) доказывающий ни делал, вероятность нераскрытия обмана (экспоненциально быстро) стремится к нулю с ростом числа раундов

Раскраска графа



Эквивалентные раскраски графа



Требуемые свойства протокола с нулевым разглашением (Zero-knowledge proof)

Полнота. Если доказываемое утверждение верно, то Доказывающий имеет возможность правильно ответить на все вопросы Проверяющего

Корректность. Если доказываемое утверждение не верно, то что бы (нечестный) доказывающий ни делал, вероятность нераскрытия обмана (экспоненциально быстро) стремится к нулю с ростом числа раундов

Нулевое разглашение. Если Доказываемое утверждение верно, то (даже нечестный) Проверяющий не узнает (почти) ничего кроме самого факта, что утверждение верно

“Почтовый” протокол раскрашиваемости графа с n вершинами

1. Доказывающий “кидает кубик” и случайным образом присваивает буквенным цветам X, Y, Z реальные цвета “красный”, “синий”, “зелёный”
2. Доказывающий берёт n сейфов, наклеивает на каждый бирку с номером от 1 до n, и кладёт в i-ый сейф шар цвета, соответствующего раскраске i-ой вершины
3. Доказывающий запирает все сейфы и посылает их Проверяющему
4. Проверяющий случайным образом выбирает ребро графа и сообщает свой выбор Доказывающему
5. Доказывающий посылает Проверяющему два ключа от сейфов, соответствующих концам выбранного ребра
6. Проверяющий открывает эти два сейфа и видит, что в них лежат шары разных цветов

Требуемые свойства протокола с нулевым разглашением (Zero-knowledge proof)

Полнота. Если доказываемое утверждение верно, то Доказывающий имеет возможность правильно ответить на все вопросы Проверяющего

Корректность. Если доказываемое утверждение не верно, то что бы (нечестный) доказывающий ни делал, вероятность нераскрытия обмана (экспоненциально быстро) стремится к нулю с ростом числа раундов

Нулевое разглашение. Если Доказываемое утверждение верно, то (даже нечестный) Проверяющий не узнает (почти) ничего кроме самого факта, что утверждение верно

Простые числа как сейфы

Проверить, что число является простым, нетрудно

Перемножить два многозначных числа нетрудно

Быстро найти раздожение большого натурального числа на простые сомножители никто пока не умеет

Нечётные простые числа бывают двух видов: $4m+1$ и $4m-1$

Красный: $\langle p, q, r, s\rangle$, $p \equiv q \equiv r \equiv s \equiv 1 (\mathrm{mod}\ 4)$

Синий: $\langle p, q, r, s\rangle$, $p \equiv q \equiv r \equiv s \equiv -1 (\mathrm{mod}\ 4)$

Зелёный: $\langle p, q, r, s\rangle$, $p \equiv q \equiv 1 (\mathrm{mod}\ 4)$, $r \equiv s \equiv -1 (\mathrm{mod}\ 4)$

Для любого цвета $pqrs \equiv 1 (\mathrm{mod}\ 4)$

“Телеграфный” протокол раскрашиваемости графа с n вершинами

1. Доказывающий “кидает кубик” и случайным образом присваивает буквенным цветам X, Y, Z реальные цвета “красный”, “синий”, “зелёный”
2. Для каждого k от 1 до n Доказывающий выбирает большие простые числа p_k, q_k, r_k, s_k, кодирующие цвет k-ой вершины
3. Доказывающий вычисляет все произведения $p_1 q_1 r_1 s_1$, ..., $p_n q_n r_n s_n$ и посылает их Проверяющему
4. Проверяющий случайным образом выбирает ребро графа и сообщает номера его концов, i и j, Доказывающему
5. Доказывающий посылает Проверяющему восемь чисел p_i, q_i, r_i, s_i, p_j, q_j, r_j, s_j
6. Проверяющий устанавливает, что эти числа – простые, и что они кодируют два разных цвета

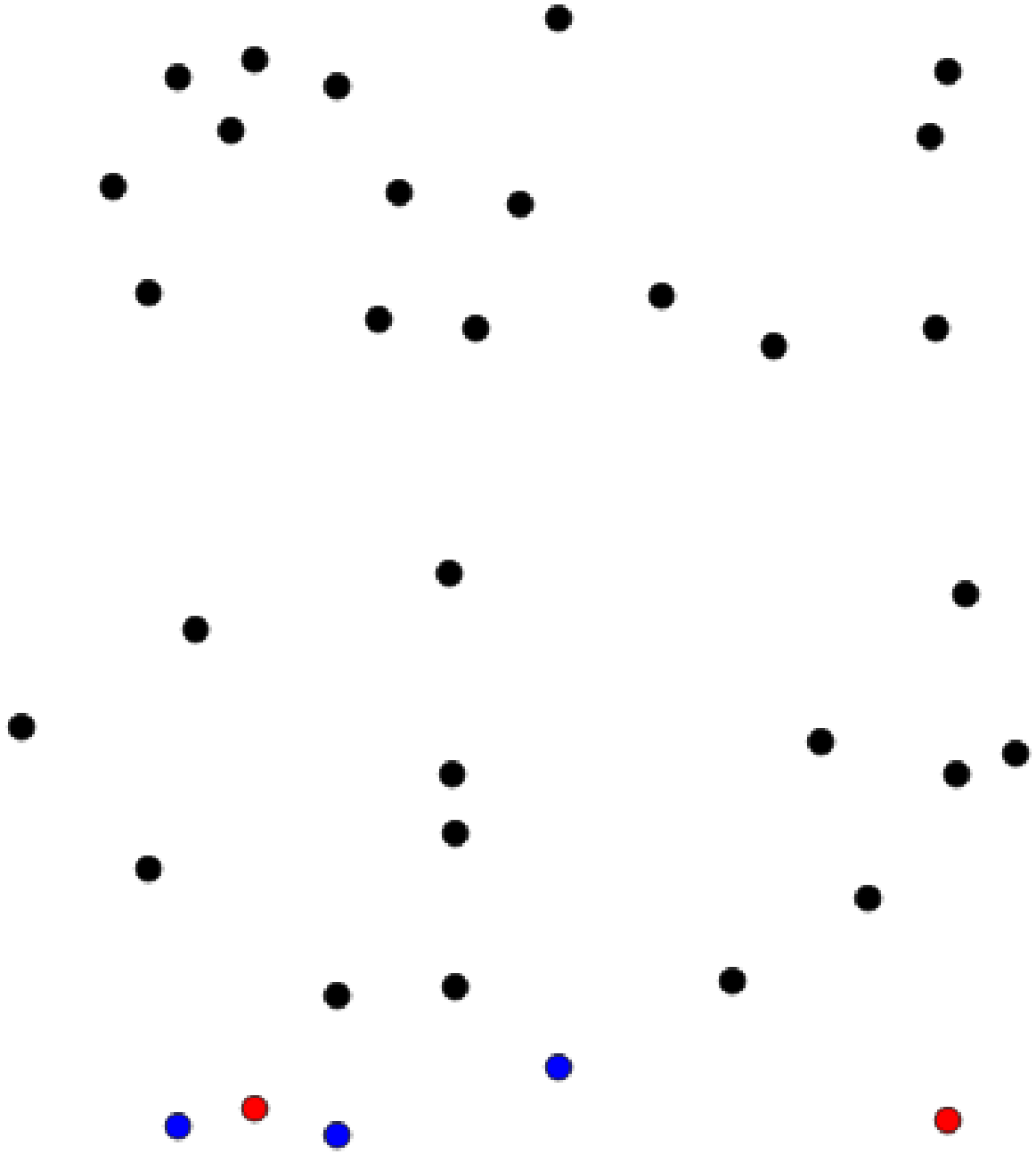
Требуемые свойства протокола с нулевым разглашением (Zero-knowledge proof)

Полнота. Если доказываемое утверждение верно, то Доказывающий имеет возможность правильно ответить на все вопросы Проверяющего

Корректность. Если доказываемое утверждение не верно, то что бы (нечестный) доказывающий ни делал, вероятность нераскрытия обмана (экспоненциально быстро) стремится к нулю с ростом числа раундов

Нулевое разглашение. Если Доказываемое утверждение верно, то (даже нечестный) Проверяющий не узнает (почти) ничего кроме самого факта, что утверждение верно

Совершенно секретный пароль



Одна голова хорошо, а две – лучше!

Посмертные записки Тяни-Толкая

“Телеграфный” протокол раскрашиваемости графа с n вершинами

1. Доказывающий “кидает кубик” и случайным образом присваивает буквенным цветам X, Y, Z реальные цвета “красный”, “синий”, “зелёный”
2. Для каждого k от 1 до n Доказывающий выбирает большие простые числа p_k, q_k, r_k, s_k, кодирующие цвет k-ой вершины
3. Доказывающий вычисляет все произведения $p_1 q_1 r_1 s_1$, . . . , $p_n q_n r_n s_n$ и посылает их Проверяющему
4. Проверяющий случайным образом выбирает ребро графа и сообщает номера его концов, i и j, Доказывающему
5. Доказывающий посылает Проверяющему восемь чисел p_i, q_i, r_i, s_i, p_j, q_j, r_j, s_j
6. Проверяющий устанавливает, что эти числа – простые, и что они кодируют два разных цвета

Посмертные записки Тяни-Толкая

$$\begin{aligned}\pi &= 3.14159265358979323846264338327950288419716939 9\ldots \\ &= 11.00100100001111110110101010001000100001011010 0\ldots\end{aligned}$$

Протокол Тяни-Толкая для графа с n вершинами и m рёбрами ($100m$ раундов)

1)Тяни-Толкай $100m$ раз "кидает кубик" и получает $100m$ случайных соответствий X, Y, Z реальным цветам

2)Тяни-Толкай Тяни-Толкай выбирает простые числа $p_{\ell,k}$, $q_{\ell,k}$, $r_{\ell,k}$, $s_{\ell,k}$, кодирующие цвет k-ой вершины в ℓ-ом раунде, $\ell = 1, \ldots, 100m$, $k = 1, \ldots, n$

3)Тяни-Толкай записывает все произведения $t_{\ell,k} = p_{\ell,k} q_{\ell,k} r_{\ell,k} s_{\ell,k}$ и из их десятичных записей образует число $t = \overline{t_{1,1}} \ldots \overline{t_{100m,n}}$

4)Тяни-Толкай отбрасывает в числе π первые t двоичных знаков, а последующие использует как датчик случайных чисел для выбора концов i_ℓ и j_ℓ рёбер, проверяемых в ℓ-ом раунде

5)Тяни-Толкай записывает числа p_{i_ℓ}, q_{i_ℓ}, r_{i_ℓ}, s_{i_ℓ}, p_{j_ℓ}, q_{j_ℓ}, r_{j_ℓ}, s_{j_ℓ}

Читателю остаётся проверить, что Тяни-Толкай следовал протоколу и что каждая такая восьмёрка кодирует пару разных цветов

Благоразумный Буратино