Задача 1. а) Пусть f — односторонняя функция, вычисляемая за время n^c. Определим g как $g(xy) = f(x)y$, где $|x| = m$, $|y| \sim m^c$ для подходящего m (под ab имеется в виду конкатенация слов a и b). Заметим, что g можно вычислить за квадратичное время от входа. Покажем, что g односторонняя. Пусть это не так, пусть существует ее обратитель R, ошибающаяся с непренебрегаемой вероятностью $1 - \frac{1}{p(n)}$ для бесконечно многих n. Тогда f тоже можно обратить. Действительно, определим ее обратитель Q как $Q(u) = R(uv)[0:k]$ для случайного v подходящей длины, т.е. Q раздувает свой аргумент справа случайным образом и кормит полученное слово обратителю R, и у результата берется префикс длины k, где k — длина аргумента f. Тогда

$$\Pr_x\left\{f\left(Q\big(f(x)\big)\right) = f(x)\right\} \geq \Pr_{xy}\left\{g\left(R\big(g(xy)\big)\right) = g(xy)\right\} \geq \frac{1}{p(|xy|)} \geq \frac{1}{p(|x|)}.$$

Полученное противоречие доказывает существование односторонней функции, вычислимой за квадрат (тем более и за куб) от длины входа.

б) Пронумеруем машины Тьюринга $M_1, M_2, \ldots$ так, чтобы длина описания n-й машины была полиномиальной от n. Пусть $M_n'(x)$ — результат работы $M_n(x)$ через $|x|^2$ шагов. Определим f как конкатенацию ответов этих усеченных машин:

$$f(x) = M_1'(x)M_2'(x) \ldots M_{|x|}'(x).$$

Она работает за время $|x|^2 \cdot |x| = O(|x|^3)$. Пусть g какая-то односторонняя функция. Тогда некоторая машина M_N из нашего списка вычисляет g. Для всех аргументов, длины больше N, $f(x)$ вычисляет $g(x)$ для N-го бита ответа. Таким образом односторонность f следует из односторонности g.

Задача 2. а) В качестве f и g возьмем ту же сильно одностороннюю функцию. Тогда $h = f = g$ — тоже сильно односторонняя.

б) Если найдутся такие односторонние функции f, g, что $f(x) \oplus g(x) = x$, то, конечно h не будет даже слабо односторонней (для данного значения y можно будет применить $\oplus$ к $y_1y_3y_5 \ldots$ и $y_2y_4y_6 \ldots$ и получить x, для которого $h(x) = y$).

Задача 3. а) Докажем, что если f и g — односторонние перестановки, то $f \circ g$ тоже односторонняя перестановка. Пусть это не так. Тогда существует эффективный алгоритм, находящий прообраз для $y = (f \circ g)(x)$, т.е. $x = (f \circ g)^{-1}(y)$ быстро вычисляется. Но ведь тогда и f легко обратить. Действительно, $f^{-1}(y) = g(x)$, где x эффективно вычисляется по нашему предположению, а $g(x)$ — по определению. Полученное противоречие доказывает, что $f \circ g$ — односторонняя. Еще $f \circ g$ — биекция, как композиция биекций, а значит и односторонняя перестановка.

Индукцией можно показать, что f^{n^c} перестановка и ее трудно обратить. Остается показать, что ее можно быстро вычислить. А это так, потому что f вычисляется за $\mathrm{poly}(n)$, а следовательно, f^{n^c} — за $n^c \cdot \mathrm{poly}(n) = \mathrm{poly}(n)$. Таким образом, f^{n^c} — односторонняя перестановка.

б) Пусть $f:\{0,1\}^n \to \{0,1\}^n$ — какая-то односторонняя функция, у которой аргумент и значение одной и той же длины (такая функция существует, если вообще односторонние функции существуют). Определим функцию $g:\{0,1\}^{2n} \to \{0,1\}^{2n}$ для $|x| = |y| = n$ как

$$g(xy) = f(y)0^n.$$

Тогда, очевидно, g — односторонняя. С другой стороны, композиция $g\big(g(x)\big) = f(0^n)0^n$ — константа и не зависит от x, а значит легко обратима.

Задача 4. Докажем сразу более сильный второй пункт. Пусть f — односторонняя функция. Пусть S — подмножество области определения f размера $\alpha(|x|)$, например множество первых $\alpha(|x|)2^{|x|}$ слов длины $|x|$, отсортированных в лексикографическом порядке (в них входит также $0^{|x|}$). Определим

$$g(x) = \begin{cases} f(x), & x \notin S \\ \;x\;, & x \in S \end{cases}.$$

Функция g удовлетворяет условиям задачи. Нужно только показать, что она односторонняя. Очевидно, g быстро вычисляется. Осталось доказать трудно обратимость. Пусть это не так и существует эффективный алгоритм R и полином p, для которых R обращает g с вероятностью $\geq 1/p(n)$ для бесконечно многих n. Заметим, что R успешно обращает $y = g(x)$ при $x \notin S$:

$$\Pr\big\{g\big(R(y)\big) = x\big\} =$$

$$= \Pr\big\{g\big(R(y)\big) = x\big|x \notin S\big\} \cdot \Pr\{x \notin S\} + \Pr\big\{g\big(R(y)\big) = x\big|x \in S\big\} \cdot \Pr\{x \in S\} \leq$$

$$\leq \Pr\big\{g\big(R(y)\big) = x\big|x \notin S\big\} + \Pr\{x \in S\} = \Pr\big\{g\big(R(y)\big) = x\big|x \notin S\big\} + \alpha(n),$$

откуда

$$\Pr\big\{g\big(R(y)\big) = x\big|x \notin S\big\} \geq \frac{1}{p(n)} - \alpha(n).$$

Покажем теперь, что f можно успешно обратить алгоритмом Q, совпадающим с R на $f(\bar{S})$ (на аргументах из $f(S)$ алгоритм Q может возвратить что угодно):

$$\Pr\left\{f\left(Q\big(f(x)\big)\right) = x\right\} \geq \Pr\left\{f\left(Q\big(f(x)\big)\right) = x\middle|x \notin S\right\} \cdot \Pr\{x \notin S\} \geq$$

$$\geq \left(\frac{1}{p(n)} - \alpha(n)\right) \cdot \big(1 - \alpha(n)\big) \sim \frac{1}{p(n)}.$$

Получили, что f можно легко обратить, вопреки определению. Значит наше предположение неверно и g — односторонняя.

Задача 6. Пронумеруем вероятностные машины Тьюринга $M_1, M_2, \ldots$. Пусть $p(.)$ — полином. В качестве Y_n берем равномерное распределение. Построим X_n. С этой целью рассмотрим следующие 2^n ($p(n)$-мерные) векторы: i-я координата вектора, соответствующего $x \in \{0,1\}^n$, равна $\Pr\{M_i(x) = 1\}$ для $i = 1, \ldots, p(n)$. Средний вектор m этих 2^n векторов лежит в их выпуклой оболочке. Значит по теореме Каратеодори из этих (экспоненциально многих) 2^n

векторов можно выбрать (полиномиальное количество) $p(n)+1$ векторов $m_1,\dots,m_{p(n)+1}$ так, чтобы m также лежал в их выпуклой оболочке, т.е. для некоторых неотрицательных α_i с суммой 1 было верно $m=\alpha_1 m_1+\dots+\alpha_{p(n)+1}m_{p(n)+1}$. Пусть вектору m_i соответствует строка $x_i\in\{0,1\}^n$. Определим X_n следующим образом: $\Pr\{X_n=x_i\}=\alpha_i$. Тогда в силу равенства

$$\Pr\{M_i(X_n)=1\}=\sum_{j=1}^{p(n)+1}\alpha_j\cdot\Pr\{M_i(x_j)=1\}=\Pr\{M_i(Y_n)=1\}$$

случайная величина X_n неотличима полиномиальными алгоритмами от равномерной Y_n. С другой стороны, эти случайные величины можно отличить схемами полиномиального размера: можно в качестве n-й схемы брать “характеристическую” схему X_n, т.е. которая выдает единицу тогда и только тогда, когда ввод является каким-то значением X_n (т.е. выдает 1 только для x_i, $i=1,\dots,p(n)+1$).

Задача 7. Пусть S — множество, на котором G и G' отличаются. Тогда для схем $\{D_n\}$ имеем

$$\Delta_n\coloneqq\left|\Pr\{D_n(G(s))=1\}-\Pr\{D_n(G'(s))=1\}\right|=$$
$$=\left|\Pr\{D_n(G(s))=1|s\in S\}-\Pr\{D_n(G'(s))=1|s\in S\}\right|\cdot\Pr\{s\in S\}.$$

В первом случае S состоит из слов с равным количеством единиц и нулей. Покажем, что G' в этом случае не является генератором псевдослучайных чисел.

По формуле Стирлинга $|S|=\binom{|s|}{|s|/2}\sim\sqrt{\frac{4}{\pi|s|}}\,2^{|s|}$, откуда $\Pr\{s\in S\}\sim\frac{C}{\sqrt{|s|}}\ge\frac{1}{p(n)}$, где $C=\frac{2}{\sqrt{\pi}}$, $p(n)=\frac{|s|}{C}$.

С другой стороны, можно считать, что величина

$$\delta_n\coloneqq\left|\Pr\{D_n(G(s))=1|s\in S\}-\Pr\{D_n(G'(s))=1|s\in S\}\right|=$$
$$=\left|\Pr\{D_n(G(s))=1|s\in S\}-p_n\right|$$

не меньше $\frac{1}{2}$ для бесконечно многих n (p_n — константа для всех s при фиксированном n). На самом деле, если это не так, то можно просто переопределить схему D_n на входе $0^{|G(s)|}$ на обратный бит. Для таким образом подобранного семейства схем и полинома $p(.)$ получим, что $\Delta_n\ge 1/2p(n)$ для бесконечно многих n, что и требовалось доказать.

Во втором случае G' является псевдослучайным генератором. Действительно,

$$|S|=\binom{|s|}{|s|/3}\sim\frac{C}{\sqrt{|s|}}\cdot\frac{3^{|s|}}{2^{2|s|/3}}$$

$$\Pr\{s\in S\}=\frac{|S|}{2^{|s|}}\sim\frac{C}{\sqrt{|s|}}\cdot e^{|s|\ln 3-\frac{5|s|}{3}\ln 2}.$$

Последнее стремится к нулю быстрее любого обратного полинома, а значит и $\Delta_n\le\Pr\{s\in S\}$ тоже. Таким образом G' — псевдослучайный генератор.

Задача 8. Пусть $H:\{0,1\}^n \to \{0,1\}^{p(n)}$, $p(n) > n$ — генератор псевдослучайных чисел. Сначала построим псевдослучайный генератор G, для которого и G' генератор. Пусть $G(xx) = H(x)H\big(H(x)\big)$ (т.е. конкатенация $H(x)$ и $H\big(H(x)\big)$), а на всех остальных аргументах (т.е. на аргументах не вида xx) G совпадает с H. Проверим, что G генератор. Для любых схем $\{D_n\}$ и равномерного x

$$\Pr\{D_n\big(G(x)\big) = 1\} =$$

$$= \Pr\{D_n\big(G(x)\big) = 1 \big| x \neq yy\} \cdot \Pr\{x \neq yy\} + \Pr\{D_n\big(G(x)\big) = 1 \big| x = yy\} \cdot \Pr\{x = yy\} =$$

$$= \Pr\{D_n\big(G(x)\big) = 1 \big| x \neq yy\} \cdot \left(1 - \frac{1}{2^{n/2}}\right) + \Pr\{D_n\big(G(x)\big) = 1 \big| x = yy\} \cdot \frac{1}{2^{n/2}}.$$

Значит, если $U_{p(n)}$ — равномерное распределение, то

$$\left|\Pr\{D_n\big(G(x)\big) = 1\} - \Pr\{D_n\big(U_{p(n)}\big) = 1\}\right| \leq$$

$$\leq \left|\Pr\{D_n\big(G(x)\big) = 1 \big| x \neq yy\} - \Pr\{D_n\big(U_{p(n)}\big) = 1 \big| x \neq yy\}\right| \cdot \left(1 - \frac{1}{2^{n/2}}\right) +$$

$$+ \left|\Pr\{D_n\big(G(x)\big) = 1 \big| x = yy\} - \Pr\{D_n\big(U_{p(n)}\big) = 1 \big| x = yy\}\right| \cdot \frac{1}{2^{n/2}} \leq$$

$$\leq \frac{1}{q(n)} \cdot \left(1 - \frac{1}{2^{n/2}}\right) + \frac{1}{2^{n/2}} \leq \frac{1}{q(n)},$$

где $q(.)$ — полином, обратная которой ограничивает успех противника для генератора H. Таким образом, G — генератор. Проверим, что $G':\{0,1\}^n \to \{0,1\}^{p(n)+p(p(n))}$ тоже генератор. Действительно, $H(x)$ распределена равномерно на $\{0,1\}^{p(n)}$ с точностью $1/q(n)$, а значит $H\big(H(x)\big)$ распределена равномерно на $\{0,1\}^{p(n)+p(p(n))}$ с точностью

$$\frac{1}{q(n)} \cdot \left(1 - \frac{1}{q(n)}\right) + \frac{1}{q(n)} < \frac{2}{q(n)}.$$

Это значит, что $G'(x) = H(x)H\big(H(x)\big)$ распределена равномерно на $\{0,1\}^{p(n)+p(p(n))}$ с точностью $3/q(n)$, что и требовалось доказать.

Теперь укажем генератор G, для которого G' не является псевдослучайным генератором. Определим $G(xx) = H(x)H(x)$ и $G(x) = H(x)$ для аргументов $x \neq yy$. Тогда, аналогично вышесказанному, G — псевдослучайный генератор ($G(x)$ отличается от $H(x)$ только на $2^{-n/2}$ доле аргументов). Но, очевидно, G' не является псевдослучайным генератором, так как значения $G'(x) = H(x)H(x)$ составляют (слишком малую) $2^{p(n)}/2^{2p(n)} = 2^{-p(n)}$ долю множества $\{0,1\}^{2p(n)}$, т.е. $G'(x)$ отличается от $U_{2p(n)}$ с точностью $1 - 2^{-p(n)}$.

Задача 1. а) Данное определение можно интерпретировать следующим образом: A — противник, $\{X_n\}_{n\in\mathbb{N}}$ — некий набор сообщений, $h(1^n, X_n)$ — любая информация про X_n, к которой противник A имеет доступ, а $f(1^n, X_n)$ — некая частная информация про X_n, в которой нуждается противник A. Неравенство в определении означает, что для любого взломщика A существует его симулятор A', который, не используя закодированное сообщение, добивается нужной информации почти с той же вероятностью, что и A'. Иными словами, для любого противника, вне зависимости из какого распределения берутся сообщения, вне зависимости к какой информации противник имеет доступ и вне зависимости от того, к какой информации он стремится, с большой вероятностью закодированное сообщение будет для него столь же полезным, сколько её длина.

б) Нужно показать, что при некотором подборе X_n, f, h, A никакой A' не сможет приближать A. Определим X_n как равномерное распределение на $\{x_n, y_n\}$, функцию f как $f(1^n, x_n) = 0$, $f(1^n, y_n) = 1$, а функцию h как независимую с f функцию, например константу. Пусть A любой алгоритм. Тогда для любого A' случайная величина $A'\big(n, h(1^n, X_n)\big)$ не зависит от $f(1^n, X_n)$, которая в свою очередь равномерно распределена на $\{0, 1\}$. В таком случае

$$\Pr\{A'\big(n, h(1^n, X_n)\big) = f(1^n, X_n)\} \leq \frac{1}{2}.$$

С другой стороны понятно, что для некоторого полинома $p(\cdot)$ верно, что $|E(1^n, X_n)| < p(n)$, но в то же время найдется $x \in \{0,1\}^{p(n)}$ со свойством $\Pr\{|E(x, X_n)| < p(n)\} < 1/2$. Т.е. как A' бы не подобрал ключ, все равно с вероятностью больше $1/2$ он не сможет симулировать поведение $A\big(n, E(1^n, X_n), h(1^n, X_n)\big)$.

в) Пусть имеется изначальное определение алгоритмами. Тогда можно моделировать семейство схем $\{C_n\}$ с помощью алгоритма A с подсказкой $\alpha(n)$. В определении A заменим на $\{C_n\}$, f не поменяем, а h заменим на $\alpha(n)$. Тогда найденный A' даст семейство схем $\{C'_n\}$.

В обратную сторону. Выберем случай, при котором вероятность максимальна. Тогда больше не будет случайности, а неравенство в определении будет верным.

Задача 3. а) Сервер должен проверить, что $g^s = g^{r+xb} = zy^b$.

б) Заметим, что нечестный клиент может ответить правильно только на один запрос b. На самом деле, пусть он смог ответить правильно разным запросам $b, b' \in \{0,1\}$ (б.о.о. $b = 1, b' = 0$), т.е. он смог предъявить s, s' такие что $g^s = zg^{bx}$ и $g^{s'} = zg^{b'x}$ (равенства имеются в виду в $\mathbb{Z}_n$). Разделив одно равенство на другое получим $g^{s-s'} = g^{x(b-b')} = g^x$, или $x = s - s'$. Отсюда следует, что не знающий x клиент может правильно ответить только одному запросу сервера, поскольку иначе, как мы показали, он в конце концов сможет вычислить x. Значит, вероятность ошибки протокола не больше $1/2$.

в, г) Докажем сразу более общий пункт г). Вспомним, что протокол с нулевым разглашением, если для любого верификатора V^* (возможно, нечестного), в нашем случае сервера, существует некий симулятор, который способен создать диалоги с тем же распределением, какое у диалогов между честным прувером (клиентом) и V^*.

Опишем симулятор. Он равномерно выбирает s и отгадывает какой запрос $b \in \{0,1\}$ будет отослан сервером V^*. На основе b он вычисляет изначальное сообщение z как $g^s y^{-b}$. Если V^*

отправит b как запрос, то симулятор будет иметь правильный ответ и получится, что он создал запрос с правильным распределением. Ясно, что симулятор будет успешным с вероятностью $1/2$, ведь b принимает только два значения.

Задача 4. От противного, пусть существует эффективный алгоритм A, обращающий одностороннюю функцию $H: \{0,1\}^* \to \{0,1\}^n$. Тогда мы можем предъявить алгоритм B, который найдет коллизию с почти той же вероятностью и той же скоростью.

Рассмотрим только входы некоторой длины $l \geq n$. Заметим, что множество $\{0,1\}^l$ распадается на следующие 2^n классов по H:

$$C_h = \{x : x \in \{0,1\}^l \wedge H(x) = h\}, \qquad h \in \{0,1\}^n.$$

В среднем, если функция ведет себя случайно, каждый класс состоит из $k = 2^{l-n}$ элементов. Но для нашей конкретней функции некоторые классы могут быть пусты в то время, как некоторые классы содержат больше элементов, чем другие.

Алгоритм B действует следующим образом:

1. Случайно берет $x \in \{0,1\}^l$ и вычисляет $h = H(x)$;
2. С помощью A обращает h: $x' = A(h)$;
3. Если $x \neq x'$, то возвращает (x, x'). Иначе возвращается к шагу 1.

Легко видеть, что на шаге 3 вероятность успеха $1 - \frac{1}{k}$ (а это довольно много; можно повторить алгоритм несколько раз, чтобы получить нужную вероятность). Действительно, выбранный на первом шаге x принадлежит к C_h с вероятностью $\Pr\{C_h\} = |C_h| 2^{-l}$. А значит, на шаге 3 вероятность неуспеха, т.е. $x = x'$, есть $1/|C_h|$, учитывая равномерность выбора x и тот факт, что A не знает какой x из C_h выбран, ведь его мы кормим только h. Таким образом, в общем вероятность неуспеха

$$\sum_{h \in \{0,1\}^n} \Pr\{C_h\} \cdot \Pr\{\text{неуспех на } C_h\} = \sum_{h \in \{0,1\}^n} \frac{|C_n|}{2^l} \cdot \frac{1}{|C_h|} = 2^{n-l},$$

который меньше любого обратного полинома. Противоречие.

Задача 5. а) Пусть противник перехватил подпись s_i под сообщением i. Тогда он сможет подделать подпись s_j под всяким сообщением $j < i$ применив $i - j$ раз f:

$$s_j = f^{(m-j)}(x) = f^{(i-j)}\left(f^{(m-i)}(x)\right) = f^{(i-j)}(s_i).$$

б) Для сообщений $j > i$ противник не сможет подделать подписи s_j с существенной вероятностью, потому что в силу равенства $s_i = f^{(j-i)}(s_j)$ для этого нужно обратить одностороннюю перестановку $f^{(j-i)}$, которое не удастся с большой вероятностью.

в) Можно в качестве подписи сообщения i брать конкатенацию $f^{(m-i)}(x)$ и $f^{(i)}(x)$, тогда как доказано выше, трудно будет обратить половину подписи. Тем самым, схема станет надежной.

Задача 7. а) Пусть $k = 2$, т.е. есть генерал G и два полковника L_1, L_2.

Случай 1: G — предатель. G может L_1 дать команду атаковать, а L_2 — отступать. Тогда если полковники честные, то они, согласно исполнительности, должны следовать команде генерала. Но, с другой стороны, по согласованности должны делать то же самое. Такое, конечно, невозможно.

Случай 2: L_2 — предатель. Рассмотрим следующий сценарий: G командует атаковать, L_2 отступает. Тогда, согласно исполнительности L_1 должен атаковать. С другой стороны, по согласованности L_1 обязан следовать за L_2 и отступать. Очевидно, такое невозможно.

б) в) Докажем сразу более общий пункт в). Рассмотрим следующий протокол $BA(n, m)$ (Byzantine agreement), где n — число командиров (не считая генерала), а m — число предателей, причем $n \geq 3m$:

Определим какое-то значение по умолчанию v_{def} (например «Атаковать»), которое заменит не присланное сообщение генерала. Определим $v = \text{maj}\{v_1, \dots, v_n\}$, где v_i — сообщение, присланное генералом полковнику i.

Протокол $\boldsymbol{BA(n, 0)}$ (нет предателей)

1. Генерал отправляет v всем полковникам;
2. Каждый полковник использует значение v_{def} в случае если не получил значения.

Протокол $\boldsymbol{BA(n, m)}$ (есть m предателей)

1. Генерал отправляет v всем полковникам;
2. i-й полковник
   - Использует значение v_i — команда, полученное от генерала (или v_{def});
   - Отправляет v_i всем остальным $n - 1$ полковникам согласно $BA(n - 1, m - 1)$;
3. Для i-го полковникова
   - Пусть v_j — значение, полученное из полковника j (или v_{def});
   - i-й полковник использует значение $\text{maj}\{v_1, \dots, v_{n-1}\}$.

Докажем корректность протокола $BA(n, m)$. Но сначала разберемся с вспомогательным утверждением:

Лемма. Для любых m, k и $n \geq 2m + k$ протокол $BA(n, m) = BA(n, m, k)$ удовлетворяет условию исполнительности, если число предателей не превосходит m.

Доказательство. Индукция по k. В случае $k = 0$ все очевидно, и генерал, и полковники честные, следовательно $BA(n, m, 0)$ сработает. Пусть $BA(n, m, k - 1)$ с $k > 0$ удовлетворяет условию исполнительности. На шаге 1 честный генерал отправляет v всем n полковникам. На шаге 2 каждый полковник применяет $BA(n, m, k - 1)$. Имеем $n \geq 2m + k$, а значит и $n - 1 \geq 2m + (k - 1) \geq 2m$. По предположению индукции каждый честный полковник получит $v_j = v$ из каждого честного полковника j. Поскольку число предателей не больше m и $n - 1 \geq 2m$, т.е. $m \leq (n - 1)/2$, то большинство полковников честные. Значит у каждого честного полковника v и есть maj от полученных значений из оставшихся n полковников, и шаг 3 удовлетворяет условию исполнительности. Лемма доказана.

Докажем теперь, что $BA(m) = BA(n, m)$ удовлетворяет условиям исполнительности и согласованности для любого m, если $n \geq 3m$ и количество предателей не больше m.

Снова индукция, но на этот раз по m. Случай $m = 0$ ясен, нет предателей и все работает как положено. Пусть $BA(m-1)$ удовлетворяет условиям исполнительности и согласованности для $m > 0$. Докажем это и для $BA(m)$.

Случай 1: генерал честный. Положив $k = m$ в лемму получим, что $BA(m)$ удовлетворяет условию исполнительности. Остается заметить, что согласованность следует из исполнительности в случае честного генерала.

Случай 2: генерал предатель. Есть не больше m предателей, а значит есть не больше $m-1$ предателей среди полковников. С другой стороны, количество полковников больше $3m - 1 > 3(m-1)$. Значит можно применить предположение индукции о том, что $BA(m-1)$ удовлетворяет исполнительности и согласованности. Следовательно, каждые два честных полковника получат то же значение v_j на шаге 3 (это следует из исполнительности, если один из полковников — j, и из согласованности — иначе). Таким образом, каждые два полковника получат один и тот же набор значений на шаге 3, а следовательно, у них совпадают maj от этих значений, доказав согласованность.