О наполнении и закупоривании бутылок

М. Л. Гервер

Все успел: обед разогрел,
Быстро съел — за уроки сел,
«Квант» почитал — в футбол
поиграл...
Милг. «Мисти-фисти, или
сказки про пострела»

Человек каждый день много дел разных делает. Идет утром в школу (если он школьник), домой вернется — обед разогреет и т. д. (см. эпиграф). Дела обычно менять местами можно: сначала в футбол поиграть — потом за уроки сесть, сперва обед съесть — потом его разогреть...

Стоп! С обедом что-то не вышло — не все на свете переставлять разрешается.

Это относится и к совсем серьезным вещам: высотному крану нечего делать на стройке, пока фундамент дома не заложен; нельзя «для сокращения сроков» пахать, сеять и молотить одновременно...

С тем, что задержка в одном деле может помешать начать другие, приходится считаться при планировании — например, когда оценивают, сколько времени потребует большой комплекс взаимосвязанных работ. В заметке на шутливом примере показано, какого рода математические задачи могут встретиться при этом.

1. Постановка задачи

Привезли на автоматическую линию бутылки, очень много: N штук. А на линии работают 2 автомата: I и II. I бутылки наполняет, II — закупо-

Таблица 1

№ бутылки	Время наполнения в секундах	Время закупоривания в секундах
1	7	10
2	9	8
3	3	1
4	2	4
5	11	6
6	5	12

ривает. Бутылки разные: с широким горлышком и с узким, высокие, пузатые — нестандартные. Про каждую бутылку известно, сколько времени уходит на ее наполнение и сколько — на ее закупоривание.

Сколько проработает автоматическая линия с момента, когда начнет наполняться первая бутылка, до момента, когда будет закупорена последняя? Это зависит от того, в каком порядке бутылки будут в автоматы попадать (например, для 6 бутылок есть $P_6=720$ таких порядков). Возникают, таким образом, два вопроса:

Как быстро определить тот порядок, при котором время работы линии будет наименьшим? И как вычислить это наименьшее время?

2. Численный пример

Начнем с примера: пусть партия из 6 бутылок характеризуется таблицей 1.

Попробуем подавать бутылки в автоматы именно в том порядке, в котором они указаны в таблице: сначала 1-ю, потом 2-ю и так до 6-й. Тогда автоматы будут работать так.

Через 7 секунд наливающий автомат наполнит 1-ю бутылку (рис. 1), через $7+9=16$ *с.* — 2-ю, через $16+3=19$ *с.* — 3-ю, через $19+2=21$ *с.* — 4-ю, через $21+11=32$ *с.* — 5-ю, через $32+5=37$ *с.* — 6-ю*).

Закупоривающий автомат покончит с 1-й бутылкой через $7+10=17$ *с.* и лишь тогда сможет заняться 2-й бутылкой (которая, тем самым, простоит наполненная целую секунду). Со 2-й бутылкой он покончит через $17+8=25$ *с.* (так что 3-я бутылка простоит в ожидании пробки уже 6 *с.*). Через $25+1=26$ *с.* будет закупорена 3-я бутылка, через $26+4=30$ *с.* — 4-я. Теперь закупоривающий автомат мог бы заняться 5-й бутылкой, но та, как мы знаем, еще не наполнена, поэтому автомат 2 *с.* простоит без дела. Через $32+6=38$ *с.* будет закупорена 5-я бутылка, а через $38+12=50$ *с.*, наконец, и 6-я.

Построим вспомогательную таблицу 2.

*) Считается, что, наполнив бутылку, автомат может, не теряя ни секунды, начать наполнять следующую.

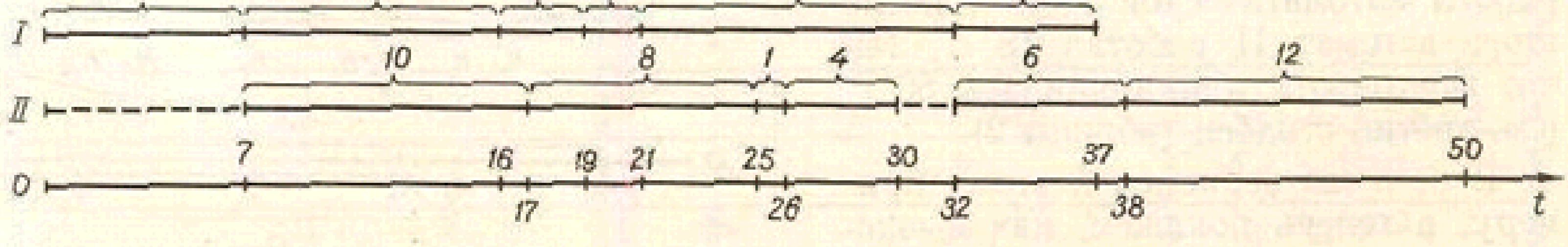

Рис. 2.

Таблица 2

7	31+10=41	7+41=48
7+ 9=16	23+ 8=31	16+31=47
16+ 3=19	22+ 1=23	19+23=42
19+ 2=21	18+ 4=22	21+22=43
21+11=32	12+ 6=18	32+18=$\boxed{50}$
32+ 5=37	12	37+12=49

Числа, стоящие в ее первом столбце, уже знакомы нам: здесь записано, через сколько секунд автомат I наполнит первую бутылку, первые две бутылки, первые три бутылки и т. д. Этот столбец получается последовательным сложением (сверху вниз) чисел, стоящих во втором столбце таблицы 1. *Второй столбец таблицы 2 удобнее сначала читать снизу вверх:* в таком порядке, чтобы получился этот столбец, складывали числа из третьего столбца таблицы 1. Третий столбец 2 — сумма первых двух ее столбцов.

Второй столбец таблицы 2 имеет следующий смысл: если бы автомат II работал *без простоев*, то он закупорил бы все бутылки за 41 *с.*; если бы он закупоривал без простоев все бутылки начиная со 2-й, то потратил бы на это 31 *с.*; на закупоривание без простоев всех бутылок, начиная с 3-й, нужно 23 *с.* и т. д.

В нашем примере автомат II работал без простоев лишь после того, как в него попала 5-я бутылка. Она была туда подана (см. 5-ю строку таблицы 2) через 32 *с.* после начала работы автоматической линии; после этого автомат II работал 18 *с.*, так что вся работа линии длилась 50 *с.* (см. третий столбец таблицы 2).

В п. 5 мы вернемся к этому примеру, а теперь покажем, как вычислить время работы автоматической линии в общем случае — при условии, что порядок подачи бутылок на линию задан.

3. Время работы автоматической линии

В нашем примере 50 *с.* — это наибольшее число в третьем столбце таблицы 2. Это не случайно: справедливо следующее

Утверждение 1. *Пусть времена наполнения и закупоривания бутылок задаются таблицей 3, и пусть бутылки попадают в автоматы именно в том порядке, в котором они указаны в этой таблице. Построим по таблице 3 вспомогательную таблицу 4 (подобно тому, как выше, по таблице 1 была построена таблица 2).*

Тогда время работы автоматической линии равно наибольшему числу в третьем столбце таблицы 4.

Доказательство. Проведем две горизонтальные прямые: I и II (рис. 2.) На прямой I отложим (сле-

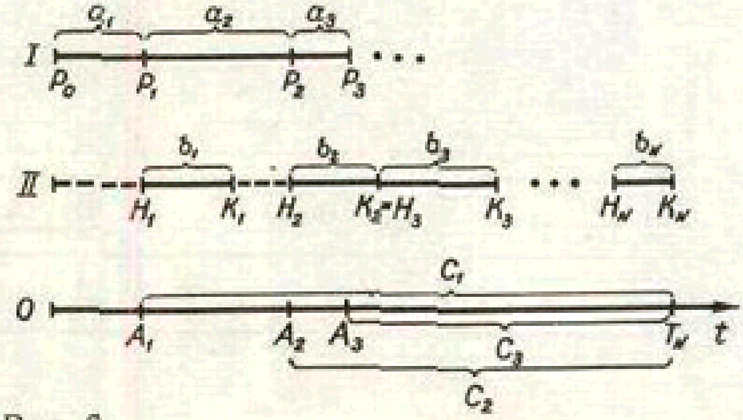

Рис. 2.

Таблица 3

№ бутылки	Время наполнения	Время закупоривания
1	a_1	b_1
2	a_2	b_2
3	a_3	b_3
...	...	...
$N-2$	a_{N-2}	b_{N-2}
$N-1$	a_{N-1}	b_{N-1}
N	a_N	b_N

ва направо) отрезки P_0P_1, $P_1P_2, \ldots$ $\ldots, P_{N-1}P_N$ (с длинами $a_1, a_2, \ldots, a_N$). На прямой II отложим (тоже слева направо) отрезки $H_1K_1, H_2K_2, \ldots, H_NK_N$ (с длинами $b_1, b_2, \ldots, b_N$). Точку H_1 расположим при этом на одной вертикали с P_1, и далее — при любом i, $1 < i \leqslant N$, — точку H_i будем располагать на одной вертикали с более правой из точек P_i и K_{i-1} (это соответствует тому, что автомат II начинает закупоривать очередную бутылку, когда она наполнена и когда закупорена предыдущая бутылка). Ортогонально спроектируем на горизонтальную ось времени t точки P_0, $P_1, \ldots, P_N$ в точки $O, A_1, \ldots, A_N$ и точку K_N в точку T_N. Ясно, что T_N — время работы автоматической линии в рассматриваемом нами случае.

Длину отрезка A_iT_N на оси t обозначим через C_i $i=1, \ldots, N$; тогда $T_N = A_i + C_i$.

Между отрезками $H_1K_1, H_2K_2, \ldots$ $\ldots, H_NK_N$ на прямой II могут быть просветы, и точка H_i располагается либо на одной вертикали с P_i, либо правее; поэтому $C_i \geqslant B_i = b_i + b_{i+1} + \ldots + b_N$. Следовательно, $T_N \geqslant A_i + B_i$ $(i = 1, \ldots, N)$.

Возьмем самый правый из просветов между отрезками H_iK_i. Правее его идут отрезки H_nK_n, $H_{n+1}K_{n+1}, \ldots, H_NK_N$, образующие один *сплошной* (без просветов) отрезок *) длины B_n. Так как H_n не совпадает с K_{n-1} (их разделяет просвет), то H_n лежит на одной вертикали с P_n и A_n. Итак, $T_N = A_n + B_n$.

Таким образом, T_N равно максимальному из чисел $A_i + B_i$ $(i = 1, \ldots$ $\ldots, N)$. Утверждение 1 доказано.

Таблица 4 строится по заданной таблице 3 чрезвычайно быстро: первые два столб-

*) При этом может случиться, что $n = N$.

Таблица 4

$a_1 = A_1$	$B_2 + b_1 = B_1$	$A_1 + B_1$
$A_1 + a_2 = A_2$	$B_3 + b_2 = B_2$	$A_2 + B_2$
$A_2 + a_3 = A_3$	$B_4 + b_3 = B_3$	$A_3 + B_3$
...	...	...
$A_{N-3} + a_{N-2} = A_{N-2}$	$B_{N-1} + b_{N-2} = B_{N-2}$	$A_{N-2} + B_{N-2}$
$A_{N-2} + a_{N-1} = A_{N-1}$	$B_N + b_{N-1} = B_{N-1}$	$A_{N-1} + B_{N-1}$
$A_{N-1} + a_N = A_N$	$b_N = B_N$	$A_N + B_N$

Таблица 5

№ бутылки	Время I	Время II
1	a	b
2	c	d

Таблица 5′

№ бутылки	Время I	Время II
2	c	d
1	a	b

ца ее получаются (каждый) за $N-1$ сложение, третий столбец — еще за N сложений (всего, таким образом, $3N-2$ сложения).

Итак, если порядок, в котором бутылки попадают на автоматическую линию, задан, то время работы линии вычисляется быстро. Остается научиться быстро определять по таблице 3 «наилучший» порядок подачи бутылок на линию, то есть такой порядок, при котором это время будет наименьшим.

4. Наилучший порядок

Сначала рассмотрим случай, когда бутылок всего две.

Пусть для одной из них времена наполнения и закупоривания равны a и b, а для другой соответственно c и d. Какой тогда порядок лучше — указанный в таблице 5 или указанный в таблице 5′?

Время работы линии в первом варианте (табл. 5) обозначим через t. В обозначениях таблицы 4 для $N=2$

$$A_1 = a, \qquad B_1 = b + a$$
$$A_2 = a + c, \quad B_2 = d.$$

Согласно утверждению 1 время t равно большему из чисел A_1+B_1 и A_2+B_2:

$$t = \max(A_1 + B_1, A_2 + B_2) = \max(a+b+d, a+c+d).$$

Обозначим сумму $a+b+c+d$ через S. Тогда формулу для t можно переписать так:

$$t = \max(S-c, S-b) = S - \min(b, c)$$

(если от S отнять меньшее из чисел b и c, то получится большее из чисел $S-b$ и $S-c$).

Аналогичная формула, разумеется, справедлива и для t', где t' — время работы линии во втором варианте (табл. 5′):

$$t' = S - \min(a, d).$$

Итак, если $\min(a, d) \geqslant \min(b, c)$, то $t \geqslant t'$.

Другими словами, при $N=2$ лучший порядок определяется по следующему правилу: выберем наименьшее из чисел a, b, c и d; если это a или d, то лучше первый вариант (табл. 5), если же это b или c, то — второй (табл. 5′). То есть при $N=2$ *в таблице, указывающей лучший порядок, наименьшее из чисел должно стоять либо в левом верхнем, либо в правом нижнем углу.*

На случай произвольного N последнее предложение обобщается так.

Утверждение 2. *«Наилучший» порядок можно определять по следующим правилам. Выберем в таблице 3 наименьшее из чисел a_1, b_1, ... , a_N, b_N*). Если это число равно a_k, то k-ю бутылку нужно подавать на линию самой первой; если же это число равно b_k, то k-ю бутылку нужно подавать на линию последней. Выяснив таким образом судьбу k-й бутылки, вычеркнем k-ю строку из таблицы 3. Среди оставшихся чисел a_i, b_i ($1 \leqslant i \leqslant N$, $i \neq k$) снова выберем наименьшее *). Если оно равно a_l, то l-ю бутылку отправим в начало очереди, а если оно равно b_l, то l-ю бутылку отправим в конец очереди (которую образуют все бутылки, кроме k-й). И так до последней бутылки.*

Мы докажем утверждение 2 в п. 6, а сейчас применим утверждения 1 и 2 к нашему примеру.

*) Если таких наименьших чисел несколько, можно взять любое из них.

Таблица 1′

№ бутылки	Время наполнения	Время закупоривания
4	2	4
6	5	12
1	7	10
2	9	8
5	11	6
3	3	1

5. Численный пример (окончание)

Наименьшее из чисел, обозначающих времена в таблице 1, равно 1. Это время закупоривания 3-й бутылки. Поэтому отправим 3-ю бутылку в самый конец очереди. Вычеркнем 3-ю строку из таблицы 1 и снова найдем наименьшее число. На этот раз оно равно 2 — это время наполнения 4-й бутылки. По нашим правилам отправляем 4-ю бутылку в начало очереди. Продолжая действовать так же, получим таблицу 1′.

Сколько будет работать линия при новом порядке? Составим, как обычно, таблицу 2′.

Максимальное число в ее 3-м столбце равно 44. Это и есть время работы линии. По сравнению со старым порядком мы выиграли 6 секунд.

Замечание: Переставим в таблице 1′ последние две строки (то есть поменяем 3-ю и 5-ю бутылки). Первые четыре *строки в таблицах 1′ и 2′ при этом не из*меняются, а последние две станут такими, как в таблицах 1″ и 2″ на странице 26.

Согласно утверждению 1 время работы линии будет по-прежнему равно 44 *с*. Таким образом, может случиться, что несколько различных порядков — «наилучшие». Утверждение 2 позволяет быстро найти один из них.

6. Доказательство утверждения 2

Выберем в таблице 3 какие-нибудь две строки, стоящие друг за другом, и поменяем их местами (см. табл. 6 и 6′).

Посмотрим, как изменится при этом таблица 4. Если мы переставим строки с номерами i и $(i+1)$, то при $k<i$ и при $k>i+1$ числа A_k и

Таблица 2′

2	37+ 4=41	2+41=43
2+ 5= 7	25+12=37	7+37=$\boxed{44}$
7+ 7=14	15+10=25	14+25=39
14+ 9=23	7+ 8=15	23+15 = 38
23+11=34	1+ 6= 7	34+ 7=41
34+ 3=37	1	37+ 1=38

Таблица 1″

3	3	1
5	11	6

Таблица 2″

23+ 3=26	6+1=7	26+7=33
26+11=37	6	37+6=43

Таблица 6

i	a	b
$i+1$	c	d

Таблица 6′

$i+1$	c	d
i	a	b

B_k не изменятся. Изменения произойдут только в строках с номерами i и $i+1$ (см. табл. 7 и 7′).

Тем самым

$$A_i+B_i=(A_{i-1}+B_{i+2})+a+b+d,$$

$$A_i'+B_i'=(A_{i-1}+B_{i+2})+b+c+d,$$

$$A_{i+1}+B_{i+1}=(A_{i-1}+B_{i+2})+a+c+d,$$

$$A_{i+1}'+B_{i+2}'=(A_{i-1}+B_{i+2})+a+b+c.$$

Обозначая $A_{i-1}+B_{i+2}$ через H и используя формулы для t и t' из п. 4, находим

$$\max(A_i+B_i,\quad A_{i+1}+B_{i+1})=H+t,$$

$$\max(A_i'+B_i',\ A_{i+1}'+B_{i+1}')=H+t'.$$

Наибольшее из чисел A_k+B_k (при $k\neq i$ и $k\neq i+1$) обозначим через K. Время работы линии в двух сравниваемых между собой вариантах обозначим через T и через T'. Тогда, согласно утверждению 1,

$$T=\max(K,\ H+t),$$

$$T'=\max(K,\ H+t').$$

Ясно, что если $t\geqslant t'$, то $T\geqslant T'$*). В п. 4 мы доказали, что $t\geqslant t'$, если $\min(a,\ d)\geqslant\min(b,\ c)$.

Поэтому только что полученный результат можно сформулировать так:

Лемма. *Выберем в таблице 3 две произвольные соседние строки (см.*

*) Отметим, что если $t>t'$, то не обязательно $T>T'$. Может случиться, что $\max(K, H+t,\ H+t')=K$ и тогда $T=T'$ независимо от того, какое из чисел t и t' больше.

Таблица 7

$A_{i-1}+a=A_i$	$B_{i+1}+b=B_i$	A_i+B_i
$A_i+c=A_{i+1}$	$B_{i+2}+d=B_{i+1}$	$A_{i+1}+B_{i+1}$

Таблица 7′

$A_{i-1}'+c=A_i'$	$B_{i+1}'+d=B_i'$	$A_i'+B_i'$
$A_i'+a=A_{i-1}'$	$B_{i+2}'+b=B_{i+1}'$	$A_{i+1}'+B_{i+1}'$

табл. 6). *Если для них*

$$\min(a, d) \geqslant \min(b, c),$$

то эти строки можно переставить (*см. табл.* 6′), *и время работы линии при этом не увеличится.*

Отсюда уже следует утверждение 2 из п. 4. Докажем его.

Начнем с двух простых замечаний.

1) Порядок, к которому мы придем, применяя правила п. 4, не зависит от того, в каком порядке бутылки были расставлены первоначально. Поэтому достаточно доказать утверждение 2 в том случае, когда порядок, указанный в таблице 3, уже был «наилучшим» (точнее, одним из «наилучших» *).

2) Допустим, что, последовательно применяя правила п. 4, мы решили (на некотором шаге) передвинуть какую-то бутылку ближе к концу очереди: с i-го места на $(i + s)$-е. Тогда это можно сделать в s этапов: сначала переставить выбранную бутылку на $(i + 1)$-е место, потом на $(i + 2)$-е и т. д. То же верно и в случае, если мы должны передвинуть какую-то бутылку ближе к началу очереди: это также можно сделать в несколько этапов, меняя местами лишь рядом стоящие бутылки. Если убедиться, что на каждом этапе применима лемма, то тем самым будет доказано, что выполнение правил п. 4 не увеличивает времени работы линии. Согласно предыдущему замечанию только в этом и осталось убедиться, чтобы доказать утверждение 2.

Случаи, когда какая-то бутылка переносится в начало или в конец очереди, рассматриваются совершенно аналогично. Рассмотрим, например, второй из них.

Пусть по правилам п. 4 решено на некотором шаге перенести какую-то бутылку с i-го на $(i + s)$-е место, и пусть на этом шаге i-я и $(i + 1)$-я строки задаются таблицей 6. Тогда число b является наименьшим среди чисел, обозначающих времена наполнения и закупоривания для всех бутылок, еще не расставленных к этому моменту на свои окончательные места. В частности, $b = \min(a, b, c, d)$. Тем самым, $\min(a, d) \geqslant \min(b, c)$. Значит, лемма применима. Точно так же доказывается, что она применима на любом этапе. Таким образом, доказательство полностью закончено.

*) См. замечание в конце п. 5.

Упражнения

1. В п. 3 мы подсчитали, что если порядок подачи бутылок на автоматическую линию задан, то время работы линии вычисляется за $3N - 2$ сложения (где N — число бутылок). А сколько действий нужно произвести для определения «наилучшего» порядка?

Если называть «действием» выбор наименьшего числа в таблице из двух столбцов и произвольного числа строк, то потребуется N действий (см. утверждение 2 в п. 4). Пока N невелико, данное определение «действия» разумно,— например, можно считать, что при составлении таблицы 1′ по таблице 1 в п. 5 мы действительно потратили N действий ($N = 6$). При больших N нахождение наименьшего числа в таблице явно перестает быть элементарным актом, так что «действием» разумнее назвать *сравнение двух чисел* (и выбор меньшего из них). Сколько таких действий-сравнений требуется для определения наилучшего порядка?

2. Можно рассматривать «задачу о бутылках» не для двух, а для n автоматов. (В математической литературе решенная в заметке задача известна под названием «задача Джонсона для $n = 2$».)

При $n > 2$ (даже при $n = 3$) эта задача до сих пор не решена. Во многом это объясняется следующим. При $n = 2$ добавление $(N + 1)$-й бутылки к N бутылкам, уже расположенным наилучшим образом, не влияет на взаимный порядок первых N бутылок. При $n > 2$ это перестает быть верным: уже при $n = N = 3$ можно так задать времена a_i, b_i, c_i ($i = 1, 2, 3$) обработки трех бутылок тремя автоматами, что «в присутствии 3-й бутылки» 1-я бутылка должна подаваться на линию раньше 2-й, а «в отсутствие 3-й бутылки», наоборот, 2-я бутылка должна подаваться на линию раньше 1-й. Приведите пример, подтверждающий это.

3. Проверьте, что при $n = 3$ бутылки, расставленные «наилучшим образом», подаются на каждый из автоматов в одном и том же порядке. При $n > 3$ это уже не обязательно так: чтобы сделать время работы линии наименьшим, может иногда понадобиться отставить в сторону очередную бутылку, обработанную некоторым автоматом A, и на следующий автомат B подать ее уже после бутылки, которая выйдет из автомата A позже этой. Приведите соответствующий пример.