

Задача 4. Монетные системы

К.А.Кноп

Условия

В России имеются в обращении монеты достоинством 1, 2, 3, 5, 10, 15, 20 и 50 копеек. Какие еще есть в обращении купюры и монеты, благодаря реформе денег, не знает никто, поэтому ограничимся этими.

Как заплатить сумму N копеек наименьшим числом монет (без сдачи)? Для этого можно использовать следующий алгоритм:

Жадный алгоритм (N)

Пусть A – наибольшая стоимость монеты, не превосходящая N . Тогда нужно взять монету A , а для нахождения остальных монет, если $A < N$, выполнить Жадный алгоритм ($N - A$)

Конец.

Например, для $N = 29$ получаем $29 = 20 + 5 + 3 + 1$. Ясно, что жадный алгоритм дает только одно из возможных решений (другое, например – $15 + 10 + 2 + 2$).

Пусть $\{a_i\}_{i=1}^M$ – монетная система. Выплату суммы N копеек монетами этой системы будем называть также разложением числа N . Разложение назовем жадным, если оно совпадает с результатом применения жадного алгоритма к числу N .

Длиной разложения будем называть количество входящих в него монет. Разложение назовем кратчайшим, если его длина – наименьшая из всех возможных длин разложений данного числа. Будем обозначать длину любого из кратчайших разложений $F(N)$, а длину жадного разложения $G(N)$.

Назовем систему удобной, если для всех N выполнено равенство $F(N) = G(N)$, и неудобной в противном случае.

4.1. Докажите, что российская монетная система – удобная.

4.2. Докажите, что если ввести в обращение монету 7 копеек, она перестанет быть удобной.

4.3. Сохранит ли система удобность, если ввести монету 9 копеек?

4.4. а) Найдите еще хотя бы одну монету, отличную от имеющихся и не большую 50, добавление которой сохраняет удобность российской системы.

б) Найти все значения N такие, что при добавлении в российскую монетную систему монеты N система остается удобной.

4.5. Докажите, что монетная система $1 = a_1 < a_2 < \dots < a_M$ является удобной, если $a_{k+1} : a_k$ для каждого $k = 1, \dots, M - 1$.

4.6. Докажите, что любая арифметическая прогрессия с первым членом, равным 1, является удобной системой.

4.7. Придумать алгоритм, проверяющий, является ли некоторая монетная система $\{a_i\}_{i=1}^M$ из M монет удобной. Алгоритм должен выполнять не более CM^3 действий, где C – константа.

Рассмотрим теперь выплаты ограниченным количеством монет (не обязательно в удобной системе): выплату числа N назовем K –выплатой, если $F(N) \leq K$. Пусть M – количество монет в монетной системе, а $N(K, M)$ – наибольшее число такое, что для всех n от 1 до $N(K, M)$ существует K –выплата числа n .

4.8. Покажите, что для российской системы ($M = 8$, монеты указаны выше) $N(3, 8) = 28$.

4.9. Существует ли (для каждого M) система из M монет, для которой $N(2, M) > 4M - 5$?

4.10. а) При каких M существует система из M монет, для которой $N(2, M) > CM$ при некотором $C > 4$?

б) Найти монетные системы из M монет, для которых $N(2, M), N(3, M)$ и т.д. ($M = 2, 3, 4, \dots$) наибольшие.

Возьмем некоторую монетную систему. Ясно, что ее “качество” зависит от того, сколько монет она требует для уплаты произвольной суммы N , и в то же время и от числа M монет в самой системе. Так, система из одной монеты (1 копейка) для суммы $N = 100$ требует 100-выплаты; а система из 100 монет (1, 2, ..., 100) для любой суммы от 1 до 100 требует всего 1-выплаты, но зато содержит $M = 100$ монет. Поэтому обе эти системы плохи. Пусть $S_1 = \max\{F(N)\}$, $N = 1, \dots, 100$, а $S_2 = M \cdot (F(1) + F(2) + \dots + F(100))$.

4.11. Найти монетную систему, минимизирующую S_1 .

4.12. Найти монетную систему, минимизирующую S_2 .

(Решившего эти задачи просим не сообщать решение в ЦБ России во

избежание новых денежных реформ).

Решения

4.1. Жадный алгоритм рекурсивно вызывает сам себя, поэтому для проверки удобства российской системы нужно только проверить, что он всегда правильно определяет первую монету. То есть в любом кратчайшем разложении числа N должна присутствовать наибольшая монета, не превосходящая N . Для проверки вручную этого свойства можно поступить следующим образом: сначала выпишем соотношения

$$\begin{array}{ll} 1 + 1 = 2 & 2 + 1 = 3 \\ 2 + 2 = 3 + 1 & 3 + 2 = 5 \\ 3 + 3 = 5 + 1 & \\ 5 + 5 = 10 & 10 + 5 = 15 \\ 10 + 10 = 20 & 15 + 5 = 20 \\ 15 + 15 = 20 + 10 & 15 + 10 = 20 + 5 \\ 20 + 20 + 10 = 50 & \\ 20 + 20 + 15 = 50 + 5 & \\ 20 + 20 + 20 = 50 + 10, & \end{array}$$

из которых следует, что если в некоторой выплате имеется более одной монеты стоимостью до 20 копеек, а также если имеются одновременно монеты 1 и 2, 2 и 3 и т.п., то эту выплату можно преобразовать таким образом, чтобы либо уменьшить количество монет в ней, либо увеличить стоимость наибольшей входящей в выплату монеты (то, что стоит слева от знаков равенства, заменить на то, что стоит справа). После того, как такую операцию проделать больше не удастся, ясно, что в выплате числа 2 есть монета 2 (так как монеты 1 не более одной); в выплате чисел 3 и 4 есть монета 3 (так как 1 и 2 не более одной и они не вместе); в выплате чисел от 5 до 9 есть монета 5 (так как из меньших монет вместе могут быть только 1 и 3, а их сумма не более 4) и т.д. Рассуждая так, получаем правильность жадного алгоритма для всех чисел.

4.2. и 4.3. В обоих случаях контрпример – 14: $F(14) = 2$, $14 = 7 + 7 = 9 + 5$, в то время как жадный алгоритм требует 3 монет: $10 + 3 + 1$, то есть $G(14) = 3$.

4.4. а) Рассуждая как и в задаче 4.1, можно показать, что единственными монетами до 50, при добавлении которых российская система остается удобной, являются 4, 6, 8, 25, 30, 35, 40 и 45.

6) Поскольку для суммы 100 существует выплата $50+50$, то и жадный алгоритм должен гарантировать 2-выплату. Отсюда следует такое необходимое условие: если добавляемая монета меньше 100, то она может быть только одной из следующих: 80 ($80=100-20$), 85 ($85=100-15$), 90 ($90=100-10$), 95, 97, 98, 99. Оказывается, что это условие является и достаточным, то есть каждую из них можно добавить. Аналогично для монет, больших 100, можно добавить монеты 110, 115, 120, 125, а также любую монету, большую 126, за исключением следующих: 131, 136, 141, 151, 152, 153, 154, 156, 161, 166, 171, 201. Эти результаты были получены на ЭВМ при помощи алгоритма проверки удобства (см. решение задачи 4.7). Алгоритмом производилась проверка для всех добавляемых монет от 4 до 300. Докажем, что можно добавить любую монету, большую 300, и система при этом останется удобной. В самом деле, пусть мы добавили монету A , $A = 300 + 50k + r = 50(6 + k) + r$, где r — остаток от деления A на 50. Уплата любой суммы, меньшей A , не изменится. Уплата любой большей суммы, не использующая монеты A , содержит не менее $6 + k$ монет, так как наибольшая из предыдущих монет равна 50. Но жадный алгоритм даст выплату, содержащую монету A , не более k монет по 50 копеек и еще не более 5 монет (наибольший остаток — 49 — требует 5-выплаты), то есть всего не более $6 + k$ монет. Поэтому система осталась удобной.

4.5. Докажем это утверждение по индукции:

База индукции. $M = 2$. Пусть имеются 2 монеты 1 и A . Любое число N может быть представлено в виде $N = xA + y$, где y — остаток от деления N на A . Для числа N жадный алгоритм дает, таким образом, выплату из $x+y$ монет, а любая другая выплата содержит $(y+zA)$ монет 1 и $(x-z)$ монет A , то есть $x+y+z(A-1)$ монет. Так как $A > 1$, то $x+y+z(A-1) > x+y$, то есть жадное разложение — кратчайшее.

Шаг индукции: пусть утверждение верно для любой системы из k монет. Рассмотрим систему $1, a_2, \dots, a_k, a_{k+1}$. Так как $a_{i+1} : a_i$, то все a_i , $i \geq 2$, делятся на a_2 . Пусть $b_i = a_i/a_2$ для каждого $i = 2, 3, \dots, k+1$. Пусть $N = xa_2 + y$. Аналогично доказательству базы индукции получаем, что в кратчайшем разложении числа N монетами 1 уплачено только число y . Для числа x и монетной системы b_2, b_3, \dots, b_{k+1} , состоящей из k монет, применим предположение индукции. Оно означает, что кратчайшая выплата числа x обеспечивается жадным алгоритмом. Значит, это же верно для числа xa_2 в монетной системе $a_2b_2, a_2b_3, \dots, a_2b_{k+1}$, которая

есть не что иное как a_2, a_3, \dots, a_{k+1} . Объединив это с выплатой числа y , получаем, что и для числа N все доказано.

4.6. Аналогично равенствам из задачи 4.1, для этой системы выписываются равенства $a_{i+1} + a_{j-1} = a_i + a_j$, “раздвигающие” выплату. Пусть имеется некоторая кратчайшая выплата числа N . Применяя эти равенства, пока это возможно, получаем, что существует выплата с тем же числом монет, в которой не более одной “промежуточной” монеты (не самой крупной и не 1). После этого заменим, если это возможно, группу из монет 1 на более крупную монету и опять, если возможно, применим “раздвижку”, и т.д. В итоге получим выплату, в которой сколько-то монет 1 (не больше, чем $a_2 - 1$), не более одной промежуточной монеты и некоторое число старших монет. Легко видеть, что жадный алгоритм дает именно эту выплату.

Участники из Канады предложили следующие две леммы, расширяющие эту задачу:

Лемма 1. Если две наибольшие монеты в удобной монетной системе равны N и $N + 1$, то она содержит все монеты от 1 до $N + 1$.

Доказательство.

Действительно, если K – наибольшая пропущенная монета, то в жадном представлении $N + 1 + K$ не менее 3 монет: $N + 1$ и не менее двух монет для представления K . В то же время оно имеет представление и из двух монет: $N + (1 + K)$. Мы пришли в противоречие с удобностью системы.

Лемма 2. Если удобная система содержит монеты N и $N + d$, $d > 0$, то она также содержит монеты $N - d$, $N - 2d$,

Доказательство.

Аналогично; очевидно, что она обобщает лемму 1.

Некоторые участники также доказали, что “фибоначчиева” система $\{1, 2, 3, 5, 8, \dots\}$ является удобной. Впрочем, этот результат, как и результаты задач 4.5 и 4.6, следует из теоремы, приведенной в решении 4.7.

4.7. Ниже будут приведены алгоритм проверки удобных монетных систем, найденный московским школьником О.Поповым, а также теорема, которая доказывает правильность этого алгоритма.

Пусть $\{a_i\}_{i=1}^M$ – монетная система. Для каждого $k = 2, \dots, M$ рас-

смотрим жадное разложение числа a_k в монетной подсистеме $\{a_i\}_{i=1}^{k-1}$:

$$a_k = \sum_{i=1}^{k-1} \beta_{k,i} a_i \quad (1)$$

Найдя коэффициенты $\beta_{k,i}$, рассмотрим числа

$$P_{k,j} = a_j + \sum_{i=j}^{k-1} \beta_{k,i} a_i \quad (j = 1, 2, \dots, k-1). \quad (2)$$

– разложения числа $P_{k,j}$, имеющие длину $1 + \sum_{i=j}^{k-1} \beta_{k,i}$.

При этом для всех допустимых значений k и j выполнено неравенство

$$a_j > \sum_{i=1}^{j-1} \beta_{k,i} a_i \quad (3)$$

– иначе разложение (1) не являлось бы жадным разложением, начиная с j -го места.

Из (1)–(3) следует, что

$$P_{k,j} > \sum_{i=1}^{k-1} \beta_{k,i} a_i = a_k,$$

таким образом, жадное разложение числа $P_{k,j}$ должно содержать монету a_k или монеты, большие ее, то есть отличаться от разложения (2). Если в некотором разложении числа N количество каждой из монет от a_j до a_{k-1} не меньше, чем их количество в (2), то такое разложение допускает преобразование, которое мы назовем $P_{k,j}$ -операцией (или просто P -операцией): заменим все эти монеты на монеты жадного разложения числа $P_{k,j}$.

Докажем, что если некоторое разложение допускает применение P -операций, то их можно сделать лишь конечное число раз. Пусть число s на 1 больше максимального из чисел $P_{k,j}$ ($k = 1, \dots, m$; $j = 1, \dots, k-1$). Если считать, что монета a_i “имеет вес” s^i , то $P_{k,j}$ -операция увеличивает суммарный вес монет в разложении: до нее вес был равен $s^j + \sum_{i=j}^{k-1} \beta_{k,i} s^i$, то есть был не больше, чем $s^j + \sum_{i=j}^{k-1} (s-1)s^i = s^j + (s^k - s^j) = s^k$, а после операции в разложении появилась либо монета a_k (с весом s^k), либо еще более тяжелая монета, а также еще какие-то монеты (так как $P_{k,j} > a_k$).

Это означает, что общий вес входящих в разложение монет увеличился. Таким образом, введенное понятие веса является полуинвариантом относительно P -операций. А поскольку для каждого N имеется лишь конечное число возможных разложений, то P -операцию можно применить лишь конечное число раз.

В итоге, когда P -операции уже невозможны, получено некоторое разложение числа N , которое мы назовем P -разложением. Наконец, мы подошли к формулировке основной теоремы — теоремы о проверке удобных систем.

Теорема. Чтобы монетная система $\{a_i\}_{i=1}^M$ являлась удобной, необходимо и достаточно, чтобы для всех $k = 2, 3, \dots, M$ и всех $j = 1, 2, \dots, k-1$ выполнялись неравенства

$$G(P_{k,j}) \leq 1 + \sum_{i=j}^{k-1} \beta_{k,i} \quad (4)$$

(Иначе говоря, чтобы длина жадного разложения каждого из чисел $P_{k,j}$ не превосходила длины его разложения (2).)

Алгоритм Попова, основанный на этой теореме (хотя сама теорема и не была им строго доказана), выполняется так: для каждой монеты a_k , $k \geq 2$, находим разложение (1). По нему строим $M(M-1)/2$ чисел $P_{k,j}$. За $C \cdot M$ действий жадного алгоритма для каждого числа $P_{k,j}$ находим длину его жадного разложения, после чего проверяем (4). Если выполнены все такие неравенства, то в силу теоремы система является удобной, если нет — неудобной. Итого алгоритм Попова требует не более $C \cdot M^3$ операций.

Доказательство.

1. Необходимость.

Если для какого-то из чисел P неравенство (4) не выполнено, то его разложение (2) является более коротким, чем жадное, то есть система неудобна по определению.

2. Достаточность.

Мы должны доказать, что при выполнении всех неравенств (4) система является удобной. Предварительно отметим, что из (4) следует, что никакая P -операция не увеличивает длины разложения. Поэтому если взять любое из кратчайших разложений любого числа и применять, пока это возможно, P -операции, придем к P -разложению этого числа, которое также будет кратчайшим.

Предположим противное: существует неудобная монетная система,

для которой, тем не менее, все неравенства (4) выполнены. Возьмем наименьшее число N , для которого жадное разложение и кратчайшее P -разложение

$$N = \sum_{i=1}^M \gamma_i a_i \quad (5)$$

не совпадают. Это означает, что для некоторых значений индексов j неравенство, аналогичное (3), неверно, то есть $a_j \leq \sum_{i=1}^{j-1} \gamma_i a_i$. Пусть r — наименьший из таких индексов. Выпишем для a_r разложение (1): $a_r = \sum_{i=1}^{r-1} \beta_{r,i} a_i$.

Будем сравнивать коэффициенты γ_i и $\beta_{r,i}$. Возможны три случая:

1) Для всех $i = 1, 2, \dots, r-1$ они попарно равны. Тогда в (5) можно заменить сумму нескольких монет $\sum_{i=1}^{j-1} \gamma_i a_i = \sum_{i=1}^{r-1} \beta_{r,i} a_i$ на одну, равную этой сумме, монету a_r , уменьшив при этом длину разложения. Это противоречит тому, что (5) — кратчайшее.

Значит, существуют значения индекса j , для которых $\gamma_j \neq \beta_{r,j}$. Пусть s — наибольший из таких индексов.

2) Если $\gamma_s \geq \beta_{r,s} + 1$, то (5) допускает $P_{r,s}$ -операцию:

$$\begin{aligned} N &= \sum_{i=1}^M \gamma_i a_i = \sum_{i=1}^{s-1} \gamma_i a_i + (\gamma_s - \beta_{r,s} - 1)a_s + (a_s + \sum_{i=s}^{r-1} \beta_{r,i} a_i) + \sum_{i=r}^M \gamma_i a_i = \\ &= \sum_{i=1}^{s-1} \gamma_i a_i + (\gamma_s - \beta_{r,s} - 1)a_s + P_{r,s} + \sum_{i=r}^M \gamma_i a_i. \end{aligned}$$

Так как в сумму входит число $P_{r,s}$, то возможна $P_{r,s}$ -операция, что противоречит тому, что (5) — P -разложение.

3) $\beta_{r,s} \geq \gamma_s + 1$. Поскольку $s < r$, в силу выбора r имеем неравенство $a_s > \sum_{i=1}^{s-1} \gamma_i a_i$. Отсюда находим

$$\begin{aligned} \sum_{i=1}^{r-1} \gamma_i a_i &= \sum_{i=1}^{s-1} \gamma_i a_i + \gamma_s a_s + \sum_{i=s+1}^{r-1} \beta_{r,i} a_i < (1 + \gamma_s) a_s + \sum_{i=s+1}^{r-1} \beta_{r,i} a_i \leq \\ &\leq \beta_{r,s} a_s + \sum_{i=s+1}^{r-1} \beta_{r,i} a_i \leq \sum_{i=1}^{r-1} \beta_{r,i} a_i = a_r, \end{aligned}$$

то есть

$$a_r > \sum_{i=1}^{r-1} \gamma_i a_i.$$

Последнее неравенство противоречит выбору r .

Полученные противоречия во всех трех случаях показывают, что предположение о том, что кратчайшее P -разложение отличается от жадного, неверно. Следовательно, жадное разложение равно одному из кратчайших разложений (а именно, P -разложению), то есть система удобна.

4.8. Это простое упражнение на усвоение введенной терминологии: нужно проверить, что кратчайшие выплаты любого числа до 28 требуют не более 3 монет, а для числа 29 трех монет не хватает. Последнее следует из задачи 4.1 и разложения $29 = 20 + 5 + 3 + 1$.

4.9. Да, существует: $1, 3, 5, \dots, 2M - 1$ (все нечетные) и последняя монета равна $2M - 2$. Рассмотрим отдельно разложения четных и нечетных чисел. Любое четное число, меньшее $2M$, выплачивается монетой 1 и еще одной монетой, любое четное число от $2M$ до $4M - 2$ выплачивается монетой $2M - 1$ и еще одной монетой, число $4M - 4$ выплачивается двумя монетами $2M - 2$. Нечетные числа до $2M - 1$ требуют одной монеты, а любое нечетное число от $2M + 1$ до $4M - 3$ выплачивается монетой $2M - 2$ и еще одной монетой. Поэтому для этой системы $N(2, M) = 4M - 4$.

4.10. а) Для четного числа монет $M = 2K$ можно рассмотреть набор из M монет

$$1, 2, 3, \dots, K, 2K + 1, 3K + 2, 4K + 3, \dots, (K + 1)K + K.$$

Двумя монетами из этого набора можно выплатить любое число до

$$(K + 1)K + 2K = (K + 3)K = M^2/4 + 3M/2.$$

Для нечетного числа $M = 2K + 1$ берем набор

$$1, 2, 3, \dots, K, K + 1, 2(K + 1) + 1, 3(K + 1) + 2, \dots, (K + 1)(K + 1) + K.$$

Двумя монетами из этого набора можно выплатить любое число до

$$\begin{aligned} (K + 1)(K + 1) + K + K + 1 &= K^2 + 3K + 2 = \\ &= (M - 1)^2/4 + 3(M - 1)/2 + 2 = M^2/4 + M + 3/4. \end{aligned}$$

В обоих случаях $N(2, M)$ имеет порядок $M^2/4$, так что $N(2, M) > C \cdot M$ для любой константы C . Легко найти оценку величины $N(2, M)$ сверху: $N(2, M) \leq (M^2 + 3M)/2$.

Докажем это:

1-выплата существует ровно для M чисел. Еще для не более чем M чисел существует 2-выплата, содержащая две равные монеты. Количество чисел, допускающих 2-выплату двумя различными монетами, не превосходит $M(M-1)/2$ — числа сочетаний из M по 2. Итого получаем $(M^2 - M)/2 + 2M = (M^2 + 3M)/2$. Если бы $N(2, M)$ было больше этой величины, то какие-то из $N(2, M)$ чисел $1, 2, \dots, M$ были бы непредставимы. Поэтому $N(2, M) \leq (M^2 + 3M)/2$. На ЭВМ участниками были исследованы монетные системы, максимизирующие значения $N(2, M)$ для значений $M \leq 10$. Эти системы приведены ниже.

$M = 3;$	$N(2, 3) = 8;$	$\{1, 3, 4\}$
$M = 4;$	$N(2, 4) = 12;$	$\{1, 3, 5, 6\}$
$M = 5;$	$N(2, 5) = 16;$	$\{1, 3, 5, 7, 8\}$
$M = 6;$	$N(2, 6) = 20;$	$\{1, 2, 5, 8, 9, 10\}$
$M = 7;$	$N(2, 7) = 26;$	$\{1, 2, 5, 8, 11, 12, 13\}$
$M = 8;$	$N(2, 8) = 32;$	$\{1, 2, 5, 8, 11, 14, 15, 16\}$
$M = 9;$	$N(2, 9) = 40;$	$\{1, 3, 4, 9, 11, 16, 17, 19, 20\}$
$M = 10;$	$N(2, 10) = 46;$	$\{1, 2, 5, 7, 11, 15, 19, 21, 22, 24\}$

Отметим, что не удалось найти никакой закономерности, которой бы подчинялись эти системы или хотя бы значения $N(2, M)$. Более того, все алгоритмы, примененные участниками для поиска таких систем на ЭВМ, по сути являлись полным перебором всех возможных систем, то есть не было найдено никакого эффективного алгоритма “отбраковки” неоптимальных систем. Все сказанное также относится и к пункту б).

Также нерешенным остался вопрос о том, для какого наибольшего значения A справедливо неравенство $N(2, M) > AM^2 + BM + C$ (выше было показано, что $1/4 \leq A \leq 1/2$).

б) Оценка сверху для $N(3, M)$ находится из тех же соображений, что и для $N(2, M)$: к $M^2 + 3M$ суммам, требующим 2-выплаты, нужно добавить еще M сумм, которые можно уплатить тремя равными монетами, $M(M-1)$ сумм, которые можно уплатить тремя монетами, ровно две из которых одинаковы, и $M(M-1)(M-2)/6$ сумм, которые можно уплатить

три разными монетами. Итого получаем

$$N(3, M) \leq \frac{M(M^2 + 15M + 2)}{6}.$$

Оценку снизу можно получить так: пусть $K = \lfloor M/3 \rfloor$, $R = M - 2K$, $R \geq K$. Рассмотрим систему: $1, 2, \dots, K, K+1, 2(K+1), 3(K+1), \dots, K(K+1), (K+1)^2+1, 2(K+1)^2+2, \dots, R(K+1)^2+R$. Три монеты этой системы можно получить все суммы до $(R+1)(K+1)^2+R \geq (K+1)^3+K = K(K^2+3K+4)$, а это величина порядка $M^3/27$. Монетные системы, максимизирующие $N(3, M)$ для небольших значений M , также были найдены при помощи ЭВМ:

$$\begin{aligned} M=3; \quad N(3, 3) &= 14; \quad \{1, 4, 6\} \\ M=4; \quad N(3, 4) &= 23; \quad \{1, 4, 7, 9\} \\ M=5; \quad N(3, 5) &= 36; \quad \{1, 4, 6, 14, 15\} \\ M=6; \quad N(3, 6) &= 52; \quad \{1, 3, 7, 9, 19, 24\} \\ M=7; \quad N(3, 7) &= 70; \quad \{1, 4, 5, 15, 18, 27, 34\} \end{aligned}$$

Покажем, как в общем случае получить для $N(K, M)$ оценку сверху. Рассмотрим, как и выше, число способов взять L монет. Оно равно числу сочетаний с повторениями из M по L , то есть $\binom{M+L-1}{L}$. Если просуммировать эти коэффициенты по $L = 1, \dots, K$, получим

$$\begin{aligned} \binom{M}{M-1} + \binom{M+1}{M-1} + \dots + \binom{M+K-1}{M-1} &= \\ &= \binom{M+K}{M} - \binom{K}{K} = \binom{M+K}{K} - 1 \end{aligned}$$

Это, как и в частных случаях, рассмотренных выше, многочлен K -й степени от M со старшим членом $1/K!$.

Участники из Канады предложили другую постановку задач 4.10–4.12: в K -выплате разрешается некоторые из K монет брать со знаком “минус”, то есть рассматривается уплата со сдачей, но общее число уплаченных и полученных в сдачу монет не больше K . Для уплат со сдачей также можно определять значения $F(N)$, $N(K, M)$, S_1 , S_2 . Например, один из полученных канадцами результатов такой: $N(3, 2) = 12$. В самом деле, пусть у нас имеются 2 монеты стоимостью A и B , $B > A$. Тогда все возможные их комбинации из не более чем трех монет будут

такими: $A, B, A+A, A+B, B+B, B-A, A+A+A, A+A+B, B+B-A, A+B+B, B+B+B, A+A-B$ (или $B-A-A$) – всего не более 12 различных комбинаций. Отсюда $N(3, 2) \leq 12$ и $B \leq 4$. Если мы хотим, чтобы все эти 12 чисел были различными, то мы должны потребовать, чтобы A не было равно $B-A$ и $A+A+A$ не было равно $B-A$. Отсюда A должно быть отлично от 1 и 2, то есть $A = 3$. Легко проверить, что для системы $\{3,4\}$ действительно $N(3, 2) = 12$.

4.11. Докажем, что $S_1 = 24$. Для этого рассмотрим систему 1, 5, 16, 40. В ней любое число до 100 требует не более чем 6-выплаты (например, $76 = 40 + 16 + 5 + 5 + 5 + 5$, $94 = 40 + 16 + 16 + 16 + 5 + 1$, $99 = 40 + 40 + 16 + 1 + 1 + 1$), поэтому $S_1 \leq 24$. С другой стороны, надо доказать, что невозможны системы, для которых $S_1 < 24$. Случаи, требующие этого доказательства, таковы:

$$\begin{aligned} S_1 = 22 &= 2 \cdot 11 = 11 \cdot 2, \\ S_1 = 21 &= 3 \cdot 7 = 7 \cdot 3, \\ S_1 = 20 &= 2 \cdot 10 = 4 \cdot 5 = 5 \cdot 4 = 10 \cdot 2, \\ S_1 = 18 &= 2 \cdot 9 = 3 \cdot 6 = 6 \cdot 3 = 9 \cdot 2, \\ S_1 = 16 &= 2 \cdot 8 = 4 \cdot 4 = 8 \cdot 2, \\ S_1 = 15 &= 3 \cdot 5 = 5 \cdot 3, \\ S_1 = 14 &= 2 \cdot 7 = 7 \cdot 2 \text{ и т.д.} \end{aligned}$$

Здесь везде первый множитель — количество монет в системе (то есть M), а второй — наибольшее требуемое для выплат число монет (то есть K).

Таким образом, нужно доказать фактически такие неравенства: $N(2, 11) < 100$, $N(3, 7) < 100$, $N(4, 5) < 100$, $N(5, 4) < 100$, $N(7, 3) < 100$, $N(11, 2) < 100$. Некоторые из них несложно доказываются без использования вычислительной техники, а для некоторых приходится снова-таки перебирать все возможные системы при помощи ЭВМ.

4.12. Докажем, что $S_2 = 3 \cdot 521 = 1563$. Рассмотрим системы $\{1, 10\}$, $\{1, 12, 19\}$, $\{1, 5, 18, 25\}$. Как показывает перебор систем из 2, 3, 4 монет, проведенный на ЭВМ, эти системы дают наилучшие значения $F(1) + F(2) + \dots + F(100)$, равные 910, 521 и 393 соответственно. Поскольку $2 \cdot 910 = 1820 > 4 \cdot 393 = 1572 > 3 \cdot 521 = 1563$, наилучшей является вторая система. Докажем, что при количестве монет более 4 значения S_2 будут больше. Пусть $M = 5$. Как показано в решении задачи 4.10, существует не более $\binom{8}{3} - 1 = 55$ чисел, требующих не более чем 3-выплаты, а остальные числа (их не менее, чем 45) требуют для выплаты не менее

4 монет. Отсюда $S_2 \geq 5 \cdot (45 \cdot 4 + 35 \cdot 3 + 15 \cdot 2 + 5 \cdot 1) = 1600$. Пусть $M = 6$. Аналогично, $S_2 \geq 6 \cdot (17 \cdot 4 + 56 \cdot 3 + 21 \cdot 2 + 6 \cdot 1) = 1704$. Для $M = 7$ получаем $S_2 \geq 7 \cdot (65 \cdot 3 + 28 \cdot 2 + 7 \cdot 1) = 7 \cdot 258 = 1806$. Для $M = 8$ получаем $S_2 \geq 8 \cdot (44 \cdot 3 + 36 \cdot 2 + 8 \cdot 1) = 1696$. При $M \geq 9$ имеется не более M чисел, требующих 1 монету и, значит, $100 - M$ чисел требуют 2 монет. $S_2 \geq M \cdot (200 - M) = 10000 - (100 - M)^2 \geq 10000 - 91^2 = 1719$. Итак, наилучшим значением M является 3, соответствующее значение $S_2 = 1563$.