

Златопольский Дмитрий Михайлович

Кандидат технических наук, доцент.

Доцент кафедры информатики и прикладной математики

Московского городского педагогического университета;

организатор и директор Музея истории

вычислительной техники.



Задания нового типа в ЕГЭ по информатике

В статье описываются методики выполнения заданий двух новых типов, представленных в проекте демонстрационного варианта Единого государственного экзамена по информатике и ИКТ 2021 года [1], который пройдет в компьютерной форме.

Задание 5

Условие

На вход алгоритма подаётся натуральное число N . Алгоритм строит по нему новое число R следующим образом.

1. Строится двоичная запись числа N .

2. К этой записи дописываются справа ещё два разряда по следующему правилу:

а) складываются все цифры двоичной записи числа N , и остаток от деления суммы на 2 дописывается в конец числа (справа). Например, запись 11100 преобразуется в запись 111001;

б) над этой записью производятся те же действия – справа дописывается остаток от деления суммы её цифр на 2.

Полученная таким образом запись (в ней на два разряда больше, чем в записи исходного числа N) является двоичной записью искомого числа R .

Укажите такое **наименьшее** число N , для которого результат ра-

боты данного алгоритма больше числа 77. В ответе это число запишите в десятичной системе счисления.

Решение

Проанализируем условие. В нем фигурируют три десятичных числа:

- 1) исходное число N ;
- 2) результат работы алгоритма R ;
- 3) некоторое целое число (в данном случае – 77), которое назовем «граничным» и обозначим G .

Согласно условию, нужно найти такое минимальное значение N , которое в результате выполнения алгоритма даст значение R , большее G .

Если соответствующие двоичные записи перечисленных чисел обозначим N_2 , R_2 и G_2 , то решение задачи сводится к двум этапам:

1) нахождение такого минимального значения N_2 , которое в результате выполнения алгоритма даст значение R_2 , большее G_2 ;

2) определение десятичного эквивалента найденного значения N_2 .

Обсудим первый этап.

Значение G_2 получить легко.

Пусть оно имеет вид (условный):

1	0	0	1	...			
---	---	---	---	-----	--	--	--

k цифр

А про значение R_2 мы знаем, что оно имеет вид:

--	--	--	--	--	--	--	--

где затененная часть совпадает со значением N_2 .

Сделаем важный для решения вывод. Длина (количество цифр) значения R_2 , соответствующего искомому N_2 , должна быть не менее длины числа G_2 . Это значит, что поиск значения N_2 можно начинать с числа, состоящего из $(k - 2)$ цифр значения G_2 .

Начнем решение.

Переведем число 77 в двоичную систему – 1001101. Согласно утверждению, сделанному чуть выше, минимально возможное значение N_2 равно 10011. Исследуем это значение:

N_2	Кол-во единиц	Двоичная запись результата (R_2)	Значение R^1
10011	Нечетное	1001110	78

Итак, значение 10011 дает нужный результат, то есть ответ: $N = 19$.

Обратим внимание на то, что в рассмотренном случае найденное число N_2 имеет вид 10011, что совпадает с укороченной на две цифры двоичной записью заданного граничного числа 77. Однако это не означает, что значение N_2 (а следовательно, и N) можно получить, отбросив две последние цифры двоичной записи заданного в условии числа G ! Подтвердим сказанное на примере.

Пусть нужно определить такое наименьшее число N , для которого результат работы описанного в условии алгоритма больше числа 90.

$$90_{10} = 1011010_2.$$

Отбросив две последние цифры, получим 10110. При этом, согласно алгоритму, результат R будет равен 90 ($1011010_2 = 90_{10}$), который не подходит по условию.

Проверка следующего числа дает искомый результат:

N_2	Кол-во единиц	R_2	R
10111	Четное	1011100	92

– то есть правильный ответ – 23 (10111₂).

Итак, алгоритм решения обсуждаемой задачи можно сформулировать следующим образом:

1) по заданному граничному числу G получить G_2 ;

2) отбросить в G_2 две последние цифры;

3) принимая новое и последующие за ним двоичные числа в качестве числа N_2 , приписывать к каждому две новые цифры по правилу:

если количество единиц в числе N_2 нечетное,

то

к нему приписываются цифры 1 и 0

иначе

к нему приписываются цифры 0 и 0

все

Первое число N_2 , дающее результат $R_2 > G_2$, определяет искомое значение N^2 .

¹ Это значение можно не определять, а сравнивать R_2 и G_2 .

² Можно доказать, что для нахождения искомого значения требуется проверка не более двух чисел.

Задания для самостоятельной работы учащихся

1. Для условия, приведенного в начале статьи, определите такое наименьшее число N , для которого результат работы данного алгоритма больше числа:

- а) 25; б) 41; в) 88.

2. На вход алгоритма подаётся натуральное число N . Алгоритм строит по нему новое число R следующим образом.

1. Строится двоичная запись числа N .

2. К этой записи дописываются справа ещё два разряда по следующему правилу:

а) если остаток от деления суммы всех цифр двоичной записи числа N на 2 равен нулю, то в конец в конец числа (справа) дописывается 1, в противном случае – дописывается 0. Например, запись 11000 преобразуется в запись 110001;

б) над новой записью производятся те же действия.

Полученная таким образом запись (в ней на два разряда больше, чем в записи исходного числа N) является двоичной записью искомого числа R .

Укажите такое наименьшее число N , для которого результат работы данного алгоритма больше числа:

- а) 21; б) 56; в) 83.

Задание 18

Условие

Квадрат разлинован на $N \times N$ клеток ($1 < N < 17$). Исполнитель Робот может перемещаться по клеткам, выполняя за одно перемещение одну из двух команд: вправо или вниз. По команде вправо Робот перемещается в соседнюю правую клетку, по команде вниз – в соседнюю нижнюю. При попытке выхода за границу квадрата Робот разрушается. Перед каждым запуском Робота в каждой клетке квадрата лежит монета достоинством от 1 до 100. Посетив клетку, Робот забирает монету с собой; это также относится к начальной и конечной клетке маршрута Робота.

Определите максимальную и минимальную денежную сумму, которую может собрать Робот, пройдя из левой верхней клетки в правую нижнюю. В ответе укажите два числа – сначала максимальную сумму, затем минимальную.

Примечание. В [1] исходные данные представляют собой электронную таблицу размером $N \times N$, каждая ячейка которой соответствует клетке квадрата.

Решение

Указанная задача является типичной задачей, решаемой методом так называемого «динамического программирования» – особого метода поиска оптимальных решений, специально приспособленного к так называемым «многошаговым», или «многоэтапным», операциям [2].

Как можно решить задачу? Очевидно, что это можно сделать по-разному.

Можно перебрать все возможные варианты перемещения Робота и выбрать тот, на котором сумма максимальна или минимальна. А можно выбирать лучший вариант шаг за шагом, на каждом этапе расчета оптимизировать только один шаг. Обычно второй способ оптимизации

оказывается проще, чем первый, особенно при большом числе шагов³.

Такая идея постепенной, пошаговой оптимизации и лежит в основе метода динамического программирования. Не вдаваясь в теоретические детали, опишем, как решается этим методом рассматриваемая задача применительно к поиску максимальной суммы. Отметим только принципиальный момент, характерный для динамического программирования, – анализ начинается с конца маршрута Робота.

Пусть квадрат, по которому перемещается Робот, состоит из 36 клеток. Квадрат будем моделировать в виде диапазона A1:F6 электронной таблицы, в ячейках которого записаны достоинства монет (см. рис. 1).

	A	B	C	D	E	F
1	13	8	20	9	20	7
2	18	9	3	10	17	14
3	8	10	16	5	2	19
4	11	9	10	4	2	5
5	15	13	13	2	7	6
6	7	4	13	5	2	3

Рис. 1

Понятно, из ячейки F5 к цели (в ячейку F6) можно попасть единственным способом – по команде вниз. При этом общая сумма, собранная Роботом, составит $6 + 3 = 9$. Конечно, расчеты удобнее проводить с помощью электронной таблицы в диапазоне A8:F13 (рис. 2).

	A	B	C	D	E	F
1	13	8	20	9	20	7
...						
6	7	4	13	5	2	3
7						
8						
9						
10						
11						
12						
13						

Рис. 2

Запишем в ячейку F13 значение 3 (или, для общего случая, формулу $=F6$), а в ячейку F12 – формулу $=F5 + F13$. При этом, естественно, на листе появятся результаты расчета по формулам (см. рис. 3).

Аналогично из всех остальных клеток крайнего правого столбца квадрата имеется только один вариант перемещения. Это значит, что можем формулу в ячейке F12 распространить (скопировать) на остальные ячейки столбца F.

Для клеток последней строки квадрата также возможен единственный вариант движения Робота (вправо). Это означает, что в ячейку E13 можно ввести формулу $=E6 + F13$, которую затем распространить (скопировать) на ячейки A13:D13.

³ При большом числе шагов в ряде случаев решение первым способом вообще не может быть реализовано. В задаче, аналогичной рассматриваемой, общее число возможных вариантов маршрутов равно числу сочетаний из $(N_1 + N_2)$ по N_1 , где N_1 – число столбцов, N_2 – число строк. При $N_1 = 10$ и $N_2 = 10$ общее число вариантов равно 184756. При увеличении N_1 и N_2 это число существенно возрастает (в [3] показано, что при $N_1 = 30$ и $N_2 = 30$ компьютер, выполняющий 1 000 000 операций в секунду, рассмотрит все возможные варианты маршрута Робота за более чем 100 000 лет!).

	A	B	C	D	E	F
...						
8						54
9						47
10						33
11						14
12						9
13	34	27	23	10	5	3

Рис. 3

Итак, мы знаем, какая сумма может собрана Роботом при движении из клеток, моделируемых на рис. 3 ячейками с уже рассчитанными значениями. В дальнейшем для удобства будем говорить о условном перемещении исполнителя по ячейкам электронной таблицы.

Сложнее с выбором решения о направлении перемещения из ячейки E12. Здесь возможные два варианта: вправо (в ячейку F12) и вниз (в ячейку E13). Какой из них лучше? Так как мы уже знаем, что из ячейки F12 до конца маршрута Робот

может собрать 9, а из ячейки E13 – 5, то понятно, что следует идти вправо. В результате сумма при движении из E12 в F13 будет равна 16. В общем случае формула для определения максимальной суммы для ячейки E12 будет такой:

$$=ЕСЛИ(F12 > E13; E5 + F12; E5 + E13)$$