Пискарев Алексей Валерьевич

ИГРА «НИМ»

ОЦЕНКА ИГРОВОЙ СИТУАЦИИ. АЛГОРИТМ ИГРЫ

Широко известна игра «Ним». Правила ее очень просты. Играют двое. На столе перед ними лежат несколько кучек каких-то предметов, например, спичек. Игроки ходят по очереди, и каждый игрок в свой ход выбирает одну из кучек и изымает из нее любое количество спичек. Если он изымет все спички из выбранной кучки, то она перестает существовать. Выигрывает тот игрок, который своим очередным ходом сумеет забрать все оставшиеся спички.

Пусть перед нами лишь одна кучка спичек. Что делать – ясно: забрать все спички из этой кучки и выиграть.

Что, если перед нами две кучки, и в них одинаковое количество спичек? Тогда мы, вероятнее всего, проиграем: сколько бы мы ни брали из одной кучки, наш партнер будет брать столько же из другой и в результате выиграет.

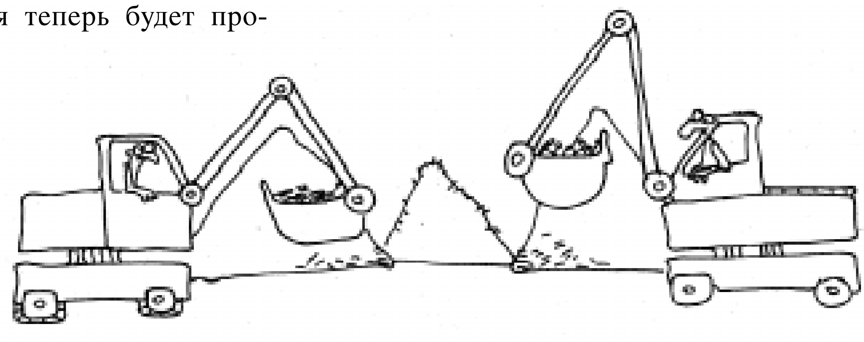
А если две кучки, но с разным количеством спичек? Тогда опять ситуация в нашу пользу: мы можем своим ходом уравнять две кучки и свести ситуацию к предыдущей, которая теперь будет проигрышной уже для нашего партнера.

Можно придумать много разных ситуаций, которые при рассмотрении оказываются заведомо выигрышными или проигрышными.

...лежат несколько кучек каких-то предметов, например, спичек...

Возникает вопрос: любая ли ситуация окажется выигрышной или проигрышной?

Выходит, что да, любая. И вот почему. Ситуация с одной-единственной спичкой заведомо выигрышная. Далее будем рассматривать последовательно все варианты более сложных ситуаций для любого количества спичек. Если в какой-то ситуации найдется такой ход, который обеспечивает противнику проигрышную ситуацию (более простую, уже рассмот-



Что если перед нами две кучки, и в них одинаковое количество спичек?

ренную), то эту ситуацию следует считать выигрышной; если же такого хода не найдется и любой ход будет переводить игру к выигрышной для противника ситуации, то такая ситуация для нас – проигрышная.

Итак, любая ситуация в игре «Спички» или выигрышная, или проигрышная. Но как определить тип сложившейся ситуации на практике, по ходу игры? Способ прямого перебора в действительности неприменим из-за своей трудоемкости. Для ответа на этот вопрос обратимся к представлению чисел в двоичной системе счисления.

Дальнейшие рассуждения сопровождаются рассмотрением примера конкретной игровой ситуации {18, 8, 14, 9, 23} – в фигурных скобках перечислены объемы имеющихся пяти кучек.

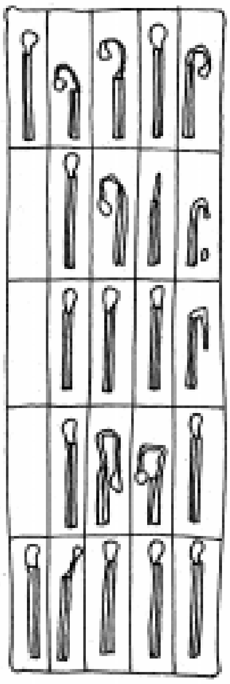
Представим все заданные числа в двоичной системе счисления:

$18=(10010)_2$
$8=(1000)_2$
$14=(1110)_2$
$9=(1001)_2$
$23=(10111)_2$

Теперь все эти представления чисел выпишем в столбик цифра под цифрой так, чтобы одна под другой оказались цифры одинаковых разрядов (то есть как при сложении в столбик):

$$\begin{array}{ccccccc}
1 & 0 & 0 & 1 & 0 & - & 18 \\
 & 1 & 0 & 0 & 0 & - & 8 \\
 & 1 & 1 & 1 & 0 & - & 14 \\
 & 1 & 0 & 0 & 1 & - & 9 \\
1 & 0 & 1 & 1 & 1 & - & 23 \\
\hline
 & \times & & \times & & &
\end{array}$$

Крестиками следует отметить те столбики, в которых стоит нечетное количество цифр 1. Теперь рассмотрим два возможных случая: какое-то количество столбиков отмечено («крестики есть») и ни один столбик не отмечен (если во всех столбиках количество цифр 1 четно – «крестиков нет»).



Представим все заданные числа в двоичной системе счисления...

Пусть «крестики есть». Тогда найдем первый слева из них и выберем любое из заданных чисел, в котором цифра над этим крестиком равна 1 (здесь таких чисел три: 8, 14 и 9; выберем, например, 8). Проделаем над числом 8 такую операцию: те цифры 1 и 0 двоичной записи этого числа, под которыми оказались крестики, заменим на противоположные (0 на 1 и 1 на 0), остальные цифры оставим без изменения:

$$\begin{array}{cccccccc}
1 & 0 & 0 & 0 & & - \; 8 & & (A) \\
\times & \downarrow & \times & \downarrow & & & & \\
\hline
0 & 0 & 1 & 0 & & - \; 2 & & (C)
\end{array}$$

Получилась двоичная запись числа 2. Следует совершить ход, оставив из 8 спичек 2 (забрать 6). Сложится новая ситуация {18, 2, 14, 9, 23}.

Покажем, что в любом случае новое число (C) будет строго меньше старого (A). Действительно, пусть есть выбранное нами двоичное число

$$A = (a_n a_{n-1} \ldots a_1 1 b_m b_{m-1} \ldots \mathrm{b}_1)_2,$$

где a_n, a_{n-1}, ..., a_1 – двоичные цифры, стоящие в столбиках левее первого слева крестика (возможно, их вовсе нет, $n = 0$), а b_n, b_{n-1}, ..., b_1 – цифры, стоящие правее первого слева крестика (их тоже может не быть, $m = 0$). После описанного выше преобразования числа A мы получим число C вида $(a_n a_{n-1} \ldots a_1 0 c_m c_{m-1} \ldots c_1)_2$. Первые n цифр останутся неизменными, следующая цифра поменяется с 1 на 0, и как-то могут поменяться остальные цифры. Из наглядного поразрядного сравнения чисел A и C:

$$A = (a_n a_{n-1} \ldots a_1 1 b_m b_{m-1} \ldots b_1)_2$$
$$C = (a_n a_{n-1} \ldots a_1 0 c_m c_{m-1} \ldots c_1)_2$$

ясно, что $A > C$ (так как $1 > 0$).

Теперь после нашего хода партнер, совершая подсчет аналогичным образом, не получит ни одного крестика (подумайте, почему). Рассмотрим этот вариант.

Итак, «крестиков нет». Тогда ход партнера не станет последним в игре. Вот почему: ход мог бы стать последним, только если на столе лежит лишь одна кучка, но в таком случае хотя бы один крестик обязательно получится. Значит, на столе осталось более одной кучки, и ближайший ход не завершит игру.

Способ прямого перебора в действительности неприменим из-за своей трудоемкости...

Пусть игрок совершит любой ход. Найдется столбик двоичных цифр, который изменится от этого хода, причем ровно на одну единицу. И тогда количество цифр 1 в этом столбике станет нечетным, и мы при подсчете поставим под ним крестик и без крестиков, таким образом, не останемся.

Понятно, что если игра будет так продолжаться (то есть если мы будем продолжать придерживаться того же метода выбора хода), то выиграем в результате именно мы, так как ни один ход нашего партнера не сможет завершить игру (рассмотрите самостоятельно, к чему приведет этот метод, если на столе останется только одна кучка спичек).

От редакции. Полезным упражнением для читателей будет написание программы для игры в «Ним», для этого можно использовать логическую операцию XOR – исключающее «ИЛИ», которая позволит простыми вычислениями определять правильные ходы.

Прочитав статью Борзых А.К. «Универсальная самообучающаяся машина из спичечных коробков» в этом же номере журнала и познакомившись с самообучающимися машинами, можно попробовать сделать программу, которая научится играть в «Ним», не зная правильной стратегии! А в качестве учителя использовать программу, которая эту стратегию знает!

НАШИ АВТОРЫ

Пискарев Алексей Валерьевич, студент V курса СПбГЭТУ (ЛЭТИ) кафедры автоматизированных систем обработки информации и управления.