Автоматы и конечно-определенные полугруппы.

И. А. Иванов -Погодаев　　　　А. Я. Белов -Канель

9 августа 2004 г.

Упражнение. а) Если Робот движется вдоль клетчатой прямой, то он не покинет некоторого луча.

б) Если же Робот гуляет по клетчатой плоскости, то он не покинет полуполосу ограниченной ширины, расположенной вдоль прямой L с рациональным углом наклона. Угловой коэффициент этой прямой равен бесконечности или есть рациональное число $\alpha = p/q$, и если p/q - несократимо, то $|p| + |q|$ не превосходит числа состояний Робота.

Под *обходом* бесконечного графа Γ мы будем понимать такое поведение Робота, что для любой вершины Γ он рано или поздно в нее попадет. Поведение одного Робота обычно тривиально и мы будем снабжать его флажками. Робот может нести с собой один или несколько флажков, оставлять флажок в вершине и проверять, находится ли рядом флажок и если да – то забирать его с собой. С помощью двух флажков Робот может обойти всю клетчатую ленту: он будет курсировать между ними и их расталкивать (см. рис.).

Приведем программу:

1. ВПЕРЕД; 2. ЕСЛИ РЯДОМ ФЛАГ ТО ВЗЯТЬ ИНАЧЕ GOTO 1; 3. ВПЕРЕД; 4. ПОЛОЖИТЬ ФЛАГ; 5. РАЗВЕРНУТЬСЯ; 6. GOTO 1;

Плоскость обходится с помощью трех флажков аналогично. Они помещаются в вершинах равнобедренного прямоугольного треугольника и расталкиваются (см.рис.)

Упражнения. а) Напишите соответствующую программу.

б) Напишите программу обхода полуплоскости для Робота с одним флагом. (Указание. см. рис.)

Интересно, что ни неогранниченную плоскость, ни ограниченную с двух сторон полосу нельзя обойти с помощью только одного флажка!

Доказательства невозможности осуществить обход используют идею виртуального автомата. Рассмотрим для примера клетчатую прямую – т.е. полосу единичной ширины. Случаи полосы ширины k, а также неограниченной плоскости аналогичны и будут предоставлены читателю. Итак, почему Робот с одним только флажком не сможет обойти всю клетчатую прямую?

Возможны два случая.

Случай 1. Робот удаляется от своего флажка сколь угодно далеко.

Случай 2. Расстояние между Роботом и флажком ограничено.

В первом случае рассуждения те же, что и в задаче про закраску только центра листа: если Робот далеко уйдет от флажка, больше чем на $10 \cdot x$, x – число состояний, то какое-то состояние повторится в процессе удаления дважды, причем к моменту повторения он удалиться от флажка. Причем в промежутке между этими повторениями он не встретит флажок. Но тогда Робот будет неограниченно удаляться от своего флажка и значит, обойдет не больше полупрямой.

Идея виртуального автомата возникает во втором случае. Если расстояние между Роботом и флажком ограничено, то система Робот–флажок имеет конечное число состояний (не больше чем число внутренних состояний Робота $\times$ максимальное расстояние). Эта система и есть виртуальный автомат. Он конечен, а для конечного автомата невозможность обойти ленту уже доказана.

Упражнения. *а) Уточните рассмотрение случая 1. Докажите, что если Робот удалится на расстояние $x+1$ от флажка, то он будет удаляться от него неограниченно. (***Указание.** *Рассмотрите положения Робота в период, заканчивающийся положением на расстоянии $x+1$ от флажка и начинающийся моментом, когда он перед этим последний раз отойдет от флажка.)*

б) Докажите, что Робот с одним флажком на неограниченной плоскости не покинет полуполосы ограниченной ширины. Докажите также, что числитель и знаменатель углового коэффициента наклона ограничены кубом числа внутренних состояний Робота.

Упражнение. *Изучается поведение Робота с двумя флажками на неограниченной плоскости. Докажте, что существует программа, позволяющая*

а) Обойти внтренность угла 90^0.

б) Посетить все внутренние клетки заданного угла ,меньшего развернутого. (Возможно, что при обходе придется выходить за его пределы.)

Доказательства невозможности осуществить обход используют идею виртуального автомата. Рассмотрим для примера клетчатую прямую – т.е. полосу единичной ширины. Случаи полосы ширины k, а также неограниченной плоскости аналогичны и будут предоставлены читателю. Итак, почему Робот с одним только флажком не сможет обойти всю клетчатую прямую?

Возможны два случая.

Случай 1. Робот удаляется от своего флажка сколь угодно далеко.

Случай 2. Расстояние между Роботом и флажком ограничено.

В первом случае рассуждения те же, что и в задаче про закраску только центра листа: если Робот далеко уйдет от флажка, больше чем на $10 \cdot x$, x – число состояний, то какое-то состояние повторится в процессе удаления дважды, причем к моменту повторения он удалиться от флажка. Причем в промежутке между этими повторениями он не встретит флажок. Но тогда Робот будет неограниченно удаляться от своего флажка и значит, обойдет не больше полупрямой.

Идея виртуального автомата возникает во втором случае. Если расстояние между Роботом и флажком ограничено, то система Робот–флажок имеет конечное число состояний (не больше чем число внутренних состояний Робота $\times$ максимальное расстояние). Эта система и есть виртуальный автомат. Он конечен, а для конечного автомата невозможность обойти ленту уже доказана.

Упражнения. *а) Уточните рассмотрение случая 1. Докажите, что если Робот удалится на расстояние $x+1$ от флажка, то он будет удаляться от него неограниченно. (***Указание.** *Рассмотрите положения Робота в период, заканчивающийся положением на расстоянии $x+1$ от флажка и начинающийся моментом, когда он перед этим последний раз отойдет от флажка.)*

б) Докажите, что Робот с одним флажком на неограниченной плоскости не покинет полуполосы ограниченной ширины. Докажите также, что числитель и знаменатель углового коэффициента наклона ограничены кубом числа внутренних состояний Робота.

Упражнение. *Изучается поведение Робота с двумя флажками на неограниченной плоскости. Докажте, что существует программа, позволяющая*

а) Обойти внтренность угла 90^0.

б) Посетить все внутренние клетки заданного угла ,меньшего развернутого. (Возможно, что при обходе придется выходить за его пределы.)

Итак, значительный кусок плоскости, почти что половину, с помощью двух флажков посетить можно! Оказывается, однако, что всю неограниченную плоскость двумя флажками не посетить. Более того, множество положений Робота и флажков за всю историю можно покрыть углом, строго меньшим развернутого в то время как ограниченную полуплоскость – т.е. внутренность развернутого угла обойти можно! И необходимый минимум для обхода плоскости – три флага. Чтобы это доказать, нужны новые идеи, которые мы извлечем из решения более простых задач.

Одна из идей состоит в том, чтобы определять состояние системы Робот–флажок–лабиринт не вообще, а в подходящий момент. Вот два примера работы этой идеи:

Задача. *На плоскости проведено 4 барьера, непроходимых для Робота, но распознаваемых им. (см. рис.) Может ли Робот все обойти?*

Докажем невозможность. Для этого достаточно заметить, что Робот бесконечно много раз должен пройти через одно из центральных полей – иначе начиная с какого–то момента он будет все время находиться только в одной четвертинке и нигде больше.

Но тогда повторится его состояние в момент прохождения этого поля и Робот будет все время топтаться на месте!

Упражнения. а) Докажите, что с двумя флажками Робот обойдет нашу область.

б) Докажите, что одного флажка мало. (**Указание.** Разберите два случая: когда флажок бесконечно много раз проносится через центр и когда он все время остается в одной из четвертинок, а через центр проходит только Робот)

Задача (А.В.Аджанс). На клетчатой ленте можно один раз записать символы. Робот их может читать, но не может ни писать, ни стирать. Можно ли расписать ленту и запрограммировать Робота так, чтобы он ее всю обошел?

Решение. Докажем, что это сделать нельзя; предположим противное. Возможны два случая. Либо Робот только конечное число раз проходит через нулевую позицию — тогда с какого-то момента он останется только в одной полупрямой. Ясно, что тогда он не обойдет всю прямую. Во втором случае Робот бесконечно много раз пройдет через нулевую позицию и тогда какие-то два его состояния в этой позиции повторятся. Далее рассуждаем как в предыдущей задаче.

Упражнение. Докажите, что один флажок здесь Роботу не поможет. (Решается аналогично предыдущему упражнению. Требуется рассмотреть проносы флажка через нулевую позицию.)

Теперь, когда мы располагаем двумя необходимыми идеями – идеей виртуального автомата и идеей состояния в нужном месте, можно приступить к доказательству невозможности обхода клетчатой плоскости спомощью только двух флажков. Мы приведем только схему разбора случаев со ссылкой на идеи, которыми эти случаи делаются. Этапы доказательства обозначены большими цифрами. Данное доказательство принадлежит А.В.Аджансу.

Значит, он бесконечное число раз далеко уходит от каждого флажка. А если Робот далеко ушел от данного флажка, то будет удаляться и дальше (см. решение задачи...) пока не наткнется на другой. Значит, далеко уйдя от одного флажка, он должен попасть к другому.

Пусть X – число его внутренних состояний.

Завершим доказательство. Итак, мы имеем режим работы Робота, как в упражнении...: Он курсирует между флажками, расположенными в полосе ширины H с данным рациональным углом наклона α. Дойдя до флажка Робот "взаимодействует"с ним и сдвигает его на ограниченное неким числом d расстояние; затем идет к другому. Сдвиги флажков устроены так, что оба все время находятся в некоторой полосе указанного вида (при этом сама ппокрывающая их полоса может только сдвигаться).

Воспользуемся идеей "повторения состояний в подходящем месте": рассмотрим состояния системы в момент нахождения Робота, скажем, у левого флажка, под "состоянием"будем понимать внутреннее состояние Робота плюс проекцию на нормаль к полосе. Пусть два состояния повторились, причем между этими повторениями других повторений нет. За это время он не более $2 \times H$ раз прошелся туда – обратно от одного флажка к другому, и, стало быть, сдвинул каждый флажок на расстояние не больше $2 \times H^2$. Мы имеем следующий режим: флажки расталкиваются и сдвигаются. Как легко видеть, положения Робота заметают угол (см.рис.).

Задача. Найдите ошибку в приведенных рассуждениях и исправьте ее.

Периодичность и остатки.

Ошибка в решении предыдущей задачи заключена в последней его части. Пусть два состояния Робота в момент подхода к первому флажку повторились. Пусть при этом повторилась проекция пары флажков на нормаль к полосе. В обоих случаях взаимодействие Робота с первым флажком а стало быть и отход ко второму флажку будут одинаковыми. Но это не значит, что и приход ко второму флажку будет таким же!

В чем же здесь дело? Пусть Робот отойдя от левого флажка и двигаясь, как нам известно, периодическим образом, пришел к правому. От чего зависит взаимодействие?

Дело в том, что встреча, вообще говоря, может произойти в любом месте периода. Значит, состояния Робота в момент встречи с правым флажком могут различаться, даже если начальные состояния в момент отхода и совпадали. Соответственно, разным может оказаться и взаимодействие Робота с правым флажком, а затем – и обратный приход к левому.

Как исправить ошибку? Поймем, когда одинаковость отхода ведет к одинаковости прихода. Это поизойдет, если путь Робота в обоих случаях отличается только на целое число периодов.

И значит, в понятие "состояние"системы следует включить еще и "остаток от вектора $\vec{F}$, соединяющего флажки по модулю периода"а затем убедиться, что число таких "остатков"конечно. Впрочем, нужно еще определить смысл слов "остаток от вектора по модулю периода". Под словом "период"в этот момент нам удобней понимать вектор $\vec{t}$, на который сдвигается Робот. [1] Этот вектор параллелен полосе. Уточним понятие "остаток". Будем вычитать из вектора $\vec{F}$ вектор $\vec{t}$, пока не получится вектор с минимальной проекцией на направление полосы. Длинна этого вектора не больше чем $|\vec{t}| + H$ (H – ширина полосы) и, значит, число таких векторов или остатков ограничено числом $4 \cdot (|\vec{t}| + H)^2$. Значит, если в понятие "состояние системы"включить еще и этот остаток, то число состояний останется ограниченным и наше доказательство невозможности обхода плоскости двумя флажками пройдет.

Следующие задачи очень похожи на только что рассмотренную:

Задача. Докажите, что бесконечное дерево степени 3 нельзя обойти

а) без флажков.

б) с помощью только одного флажка.

в) С помощью только двух флажков.

В дальнейшем мы увидим как с помощью трех флажков обойти это дерево.

Задача. С помощью одного только флажка нельзя обойти полупространство. Скажем больше: множество всех положений Робота можно покрыть конечным числом полос (полоса – это H – окрестность плоскости), имеющих общую прямую, вдоль которой движется флажок. (ср. с упражнением ...)

В конце решения задачи про два флажка на плоскости нам встретились остатки. Постараемся осознать, с чем мы столкнулись. В начале книги мы изучали остатки с помощью идеи периодичности. Здесь же мы наоборот, изучаем периодичность с помощью идеи остатка! Остаток выступил не как математическое понятие, а как идея. (Заметим, что остаток в задаче про флажки – это не совсем тот остаток, к которому мы привыкли. В процессе решения он возник не как четкое понятие, и только потом был формализован.)

Роль идеи остатка при изучении периодичности заключается в том, что состояние процесса в момент времени t управляется остатком от деления t на величину периода.

Применяя это соображение мы видим, что если рассматривать несколько процессов как один, то период общего процесса есть наименьшее общее кратное периодов составляющих процессов (а предпериод равен максимуму предпериодов). Как видите и более сложное понятие наименьшего общего кратного может выступать как идея.

На плоскости движение Робота прямолинейно и равномерно. На полуплоскости его движение тоже не сложно: "с высоты птичьего полета"будет наблюдаться "равномерный"подход к границе (если только он к ней подходит), взаимодействие с ней ("рассеивание") и затем равномерный отход от границы (возможно, в другом режиме) либо Робот будет путешествовать вдоль границы (он будет "захвачен"ею), и его движение будет опять-таки равномерным.

А как движется Робот по четвертинке плоскости? Нет ли и тут периодичности, подобной той, которую мы уже наблюдали в задаче про два флажка: не будут ли его состояния при ударе, скажем, о нижний край, периодически повторяться? Попробуем это доказать, рассуждая похожим образом.

Если Робот не взаимодействует с краями, то его движение ничем не отличается от движения по неограниченной плоскости; если Робот только конечное число раз ударяется об один из краев, то его движение в конце ничем не отличается от движения по полуплоскости. Итак, достаточно изучить случай, когда Робот бесконечное число раз ударяется об оба края. Докажем периодичность смены состояний Робота при ударе о нижний край листа.

Рассмотрим его состояния в момент соударения с нижним краем. Ясно, что два состояния повторятся. Для повторения состояний Робота при следующем ударе, надо чтобы приход к другому краю и взаимодействие осуществились тем же местом периода, как мы это уже видели при разборе задачи про два флажка. Чтобы это обеспечить, включим в понятие "состояние"еще и остаток от координаты Робота от деления на число N, которое мы выберем так: возьмем все режимы движения Робота, и пусть (x_i, y_i) – соответствующие векторы сдвига. И пускай $N = \prod_i x_i! y_i!$. Теперь-то

[1] а не то, что мы понимали до сих пор и будем понимать после этого места – повторяющийся кусок последовательности состояний.

уж заведомо, если повторились еще и остатки по модулю N, то осуществится не только одинаковый отход от нижнего края, но и одинаковый приход к левому краю и наоборот. Итак, периодичность поведения Робота – т.е. смены его состояний возле нижнего края, установлена!

Задача. Найдите ошибку в предыдущих рассуждениях. (Движения Робота могут быть и непериодичными.)

Когда я работал над Чертежником вместе с М.В.Сапиром, он сообщил мне

Удивительный факт. *Пусть $A_1, A_2, \dots,$ - последовательность натуральных чисел, и существует программа, по которой ЭВМ (с бесконечной памятью) для каждого n вычисляет A_n. Тогда существует программа для Чертежника, по которой он для любого n, начав из положения на рис.3 при $K = 2^n$, закрасит полоску из 2^{A_n} клеток, если лист бумаги - четверть плоскости.*

Для умеющих возводить в степень и логарифмировать Чертежник может заменить компьютер. (Это значит, говоря на Строгом Языке Логиков, что с его помощью можно вычислять любую Вычислимую Функцию $A(N)$.) Из этого факта можно извлечь алгоритмическую неразрешимость ряда утверждений - например, уже упомянутую теорему Матиясевича.

Таким образом, Чертежника, который совсем "не знаком"с числами, можно научить "вычислять"все, что может вычислить ЭВМ с бесконечной памятью, например, цифры в десятичном разложении корня из двух (подробнее об алгоритмах см. книгу В. А. Успенского "Машина Поста").

Однако у меня поначалу возникло сомнение в "удивительном факте"и я попытался его опровергнуть. Рассуждения были такими: докажем, к примеру, невозможность перехода от 2^n к 2^{n^2} или, еще проще, перехода от 2^n к 3^n. Если число n - очень большое, то, в процессе перехода Чертежнику придется очень много раз удариться о край (это действительно так. Советуем доказать), а дальше последовало уже описанное рассуждение о периодичности поведения Чертежника на четвертинке плоскости. Это рассуждение, вместе с найденной в нем ошибкой дало возможность понять и доказать "удивительный факт". Поэтому постарайтесь сами ее найти.

Так все-таки в чем же ошибка?

Постараемся разобраться. Пусть два состояния Робота на нижнем краю повторились, x_1 и x_2 – соответствующая координата, причем по условию $x_1 \equiv x_2 (mod N)$. Когда Робот перейдет на левый край, соответствующие координаты будут y_1 и y_2. Приход к левому краю произойдет тем же местом периода, соответственно одинаковым будет и взаимодействие. Однако может получится так, что $y_1 \not\equiv y_2 (mod N)$. И после нового прихода к нижнему краю не только $x'_1 \not\equiv x'_2 (mod N)$, но, более того, в момент следующего прихода к нижнему краю могут различится и внутренние состояния Робота, и, соответственно, разным оказаться взаимодействие с краем. Здесь и заключена ошибка.

Посмотрим на примере что происходит, и заодно постараемся осуществить переход $2^n \to 3^n$. Пусть Робот идет три раза вверх и два раза влево. Тогда если x – его начальная координата внизу, то его координата у после прихода на левую сторону угла будет равна $3/2{*}x$. Пусть затем Робот возвращается к оси OX под углом в 45^o; тогда $x' = y = 3/2x$ и $(x_1 - x'_1) = 3/2 * (x_2 - x'_2)$. Так что если x_1 и x_2 и были сравнимы помодулю N, то сравнимость x'_1 и x'_2 может нарушится, более того, расстояние между $x_1^{(k)}$ и $x_2^{(k)}$ может стать нечетным. После этого приход Робота к краю осуществится другим местом периода и подолжать вести себя тем же образом Робот не сможет.

Теперь попробуем осуществить переход $2^n \to 3^n$. Если вначале Робот находится на расстоянии $x = 2^n$ от вершины угла, то $x' = 3/2 \cdot x = 3 \cdot 2^{n-1}, x'' = 3^2 2^{n-2}, \dots, x^{(n)} = 3^n$. Однако для осуществления перехода нам надо остановить Робота на расстоянии 3^n от начала. Как это сделать? Надо на обратном пути к оси OX всякий раз проверять четность, для чего считать число тактов по модулю 2. Под тактом следует понимать пару сдвижений: на одну клетку вправо и на одну клетку вниз. Для слежением за тактами Роботу хватит одной дополнительной ячейки памяти. Итак, разобравшись с ошибкой мы осуществили переход $2^n \to 3^n$.

Исследуем поведение Робота на четвертинке. Теперь ясно, как для взаимно простых q_1 и q_2 создать программу для перехода Робота $q_1^n \to q_2^n$, программу универсальную по n. Надо действовать аналогично, сдвигаясь на q_2 полей вверх и на q_1 полей влево. Тогда координата у прихода на левый край с координатой x отхода от нижнего края будет связана равенством $y = x \cdot q_2/q_1$. Возвращаясь под углом в 45^o к левому краю, следует проверять число тактов по модулю q_1. Число состояний Робота будет, разумеется, зависеть от q_1 и q_2, но не от n. А что если q_1 и q_2 не взаимно просты? Надо взять q_3, взаимно простое с q_1, q_2 и осуществить переход в два этапа:

$$q_1^n \to q_3^n \to q_2^n$$

Можно осуществить и более сложные переходы.

Задача. Осуществить переход $2^n \to 2^{2^n}$ универсально по n.

Схема перехода.

Как видите, поведение Робота на четвертинке плоскости может быть очень сложным; "удивительный факт"означает, что оно может имитировать работу компьютера с неограниченной памятью. Постараемся во всем этом разобраться.

Простые числа играют роль регистров, а показатель степени при данном простом – содержимого соответствующего регистра. Нас интересуют прежде всего положения Робота на нижнем краю. Такое положение Робота на расстоянии $2^a 3^b 5^c 7^d$ от вершины угла, к примеру, мы рассматриваем как четыре числа a, b, c, d, которые записаны в регистры, помеченные символами $2, 3, 5, 7$. Эта кодировка называется Геделевой нумерацией. Обозначать регистр простого p мы будем $\mathfrak{P}$.

Заметьте, что всего лишь из двух регистров – координат x и y Робота мы научились извлекать любое, наперед заданное число регистров!

Что можно делать с регистрами? Движение вдоль прямой с угловым коэффициентом наклона $-p$ или $-1/p$ дает возможность увеличивать или уменьшать содержимое регистра $\mathfrak{P}$ на единичку. Проверка делимости расстояния от края на p означает проверку того, записан ли в регистре $\mathfrak{P}$ ноль или нет. Содержимое одного регистра $\mathfrak{P}_0$ мы можем, как это было подробно показано для двойки и тройки, перегнать в другой регистр $\mathfrak{P}_1$ и, более того, сразу в несколько регистров $\mathfrak{P}_1$, $\mathfrak{P}_2$,..., $\mathfrak{P}_n$. Для этого достаточно повторять, пока это возможно, следующее действие: вычитание единички из $\mathfrak{P}_0$ с последующим добавлением по единичке в каждый регистр $\mathfrak{P}_i$. С помощью этих операций содержимое каждого регистра можно увеличивать в несколько раз, копировать в другой регистр ($p_0^n \to p_2^n \to p_0^n p_1^n$), складывать содержимое двух регистров ($p_0^n p_1^n \to p_0^n p_1^m p_3^n \to p_0^n p_1^m p_3^{n+m}$), а также сбрасывать содержимое регистра ($p^n M \to M$).

Обратимся снова к схеме 1. Как осуществлялся переход $2^n \to 2^{2^n}$? После перехода $2^n \to 2 \cdot 3^n$ мы n раз удваивали содержимое регистра 2. Это было организовано с помощью регистра 3, выступившего в качестве счетчика: после каждого удвоения регистра 2 вычиталась единичка из регистра 3, и делалось это пока возможно. Мы осуществили цикл с $i = n, \dots, 0$. В общем случае, пусть мы можем сделать некоторую операцию, не затрагивающую регистр $\mathfrak{P}$, в котором записано число n. Тогда эту операцию можно повторить n раз: завершая ее, вычитаем, если это возможно, из регистра $\mathfrak{P}$ единичку и снова возобновляем. Поскольку $a \times b = a + \dots + a(b\text{раз})$, и $a^b = a \times a \times \dots \times a(b\text{раз})$, то можно умножать содержимое двух регистров и возводить в степень. Кроме того, содержимое двух регистров можно сравнивать ($p_1^n p_2^m \to p_1^n p_2^m p_3^n p_4^m$, далее вычитать, пока можно, по единичке из регистров $\mathfrak{P}_3$и $\mathfrak{P}_4$, и посмотреть, в каком из них впервые появится ноль. Потом содержимое буферных регистров $\mathfrak{P}_3$ и $\mathfrak{P}_4$ можно сбросить.

Если мы умеем вычислять функцию $f(x)$, с помощью сравнения содержимого регистров, можно вычислять и обратную функцию $f^{-1}(y)$: будем последовательно вычислять $f(1), f(2), \dots$, пока при некотором x $f(x)$ не окажется равным y.

Следующий переход понадобится при обходе дерева с тремя флажками:

$2^x 3^y 5^z \to 2^x 3^{[x/3]} 5^{3z+\bar{y}}$; где $\bar{y} \in \{0, 1, 2\}$ - остаток числа y по модулю 3.

Приведем примеры объектов, эквивалентных Роботу на четвертинке плоскости:

1. (М.В.Сапир) Имеется две неограниченных справа проволочки с костяшками счет и конечный автомат, который может перебрасывать каждую костяшку налево-направо и проверять, кончились слева костяшки, или нет.

2. Конечный автомат, снабженный двумя счетчиками. Счетчик – это регистр внешней памяти, к содержимому которого можно прибавлять или вычитать единичку, а также проверять, записан 0 или не 0. Содержимым счетчика может быть только натуральное число.

3. Робот на прямой, снабженный тремя флажками.

4. (М.В.Сапир) Робот на четвертинке плоскости, который может двигаться только под углом 45^o к осям, на две клетки вверх и одну влево, на три клетки вверх и одну влево либо наоборот. По прежнему разрешена проверка наличия края.

Что понимается под словами *эти объекты эквивалентны Роботу на четвертинке плоскости*? Поясним это на третьем примере. Каждой программе для Робота на четвертинке плоскости соответствует программа для Робота на прямой следующим образом. Роль координат x и y будут играть расстояния между левым и средним и между правым и средним флажками. Роль проверки условия "ВПЕРЕДИ КРАЙ"будет играть проверка того, находятся ли два флажка в одном месте. Вместо перемещения по координате x или y мы будем двигаться до соответствующего флажка и его сдвигать. Сказанное означает, что система "Робот на четвертинке плоскости"интерпретируется системой "Робот на прямой с тремя флажками".

Упражнение. Постройте аналогичную интерпретацию объектами 1, 2, 4.

Мы рассмотрели подробнее третий объект, поскольку он нам понадобится для обхода лабиринтов. Общая идея такова: ставятся три флажка в ряд и организуется Машина Минского – т.е. компьютер с еограниченной памятью, для которого и пишется программа обхода лабиринта. При этом следует проследить за тем чтобы не потерять память – в прямом и переносном смысле – т.е. надо двигаться вместе со своими флажками сохраняя при этом информацию, заключенную в расстояниях. Рассмотрим простейший пример – обход плоскости с помощью трех флажков. Конечно, это делается гораздо проще, но нам интересен только пример рассуждений. Расположим вначале флажки вдоль оси OX так, чтобы расстояние между средним и левым было бы равно 0, а расстояние между средним и правым – скажем, восьми, т.е. 2^3. Организуем, как обычно, регистры 2 и 3. Перегоняя единичку из регистра 2 будем подниматься на 1 шаг вверх вместе с флажками. Это осуществляется так: доходим до самого левого, передвигаем его на 1 шаг вверх, возвращаемся вниз и идем направо, проделывая то же с двумя другими флажками. (Дойдя до последнего, разворачиваемся.)

Итак, будем подниматься вместе с флажками, пока содержимое 2 не перегонится в 3. Затем, перегоняя содержимое 3 в содержимое 2 и опускаясь, вернемся в исходное положение – на ось OX. Мы обошли область вверху этой оси. Область под ней обходится аналогично, после чего к содержимому 2 добавляем единичку и повторяем весь процесс.

Для обхода трехмерного пространства можно воспользоваться регистрами 2 и 3 для движения вдоль оси OY, а регистрами 5 и 7 — для движения вдоль оси OZ. Всякий раз, перегоняя единичку из 5 в 7, мы обходим кусок слоя с помощью регистров 2 и 3 и смещаемся вдоль оси OZ нужным образом.

Упражнение. Нарисуйте схему программы для обхода Роботом с тремя флажками k-мерной решетки. Число его состояний может зависеть от k.

Займемся теперь более интересным обходом свободного дерева степени, скажем, 3 с помощью трех флажков. (см. рис.)

Вначале покажем, как обходит это дерево компьютер с неограниченной памятью. Пусть O – начальная вершина. Программа обхода состоит в переборе всех конечных путей, выходящих из O. Проходя каждый путь, компьютер возвращается назад в точку O и переходит к прохождению нового пути. Каждый путь кодируется последовательностью цифр $0, 1, 2$ которая записывается троичным разложением одного числа X. 0 соответствует команде "НАЗАД", 1 – "НАЛЕВО", 2 – "НАПРАВО". Переход к рассмотрению нового пути соответствует добавлению единички к числу X. Чтобы вернутся обратно нам надо помнить в процессе движения также число Y, кодирующее путь, обратный к пройденному отрезку пути, и число Z – кодирующее оставшуюся часть пути. После каждого смещения числа Y и Z изменяются так: $Z' = [Z/3]$, $Y' = 3Y + Z$, где $Z \in \{0, 1, 2\}$ – остаток от Z по модулю 3. (При обратном движении с помощью информации, закодированной в числе Y, 1 соответствует повороту "НАПРАВО", а 2 – "НАЛЕВО".) Пункт в) упражнения ... позволяет осуществить этот алгоритм на машине Минского.

Но как реализовать саму машину Минского на дереве с помощью трех флажков? Поставим флажки в ряд, если идти по "правилу правой руки" – т.е. все время поворачивать направо. Это дает возможность реализовать машину Минского, однако как двигаться, чтобы не впасть в беспамятство? Как не потерять флажки и информацию, связанную с ними в процессе движения? Идея такого смещения показана на рисунке:

Упражнение. Нарисуйте схему программы обхода свободного дерева степени k.

Замечание. Робот, снабженный датчиком случайных чисел с помощью всего лишь двух флажков может обойти решетку любой размерности. И это несмотря на то, что без такого датчика ему не обойти даже плоской решетки! Идея состоит в организации Машины Минского в плоскости, наральлельной плоскости XOY. Пусть один из флажков это вовсе и не флажок, а прожектор, который светит вдоль осей OX и OY. В этом случае Машина Минского организуется как и при движении Робота на четвертинке листа – попадание под луч соответствует удару о край. Чтобы обойтись без прожектра, достаточно уметь определять взаимное расположение двух флажков. Для этого достаточно пройти от одного к другому монотонным путем (т.е. не содержащим движений в противоположных направлениях). Этого добиваемся так: пусть Робот блуждает вдоль плоскости, где расположены флажки. Рано или поздно он отойдет от первого и придет ко второму флажку, в силу теоремы о возвращении. Будем наблюдать такие прохождения не трогая флажки, пока Робот не пройдет от одного к другому монотонным путем. Поскольку возвращение к исходной точке при случайном блуждании по плоскости происходит неограниченное число раз, такое явление неизбежно, что и позволяет симитировать прожектор.

Лабиринт, который не обойти.

Итак, Машина Минского равносильна компьютеру с бесконечными ресурсами. Нам удавалось организовать ее в довольно сложных ситуациях, например, на дереве. Создается впечатление, что если есть много флажков, а тем более взаимодействующих Роботов, то можно обойти все, что угодно. Однако это не так. Справедлива

Теорема. *Существует бесконечный связный граф, такой, что любое конечное семейство из флажков и взаимодействующих Роботов не сможет покинуть конечной области.*

Чтобы привести пример такого графа, нам потребуется ввести несколько новых понятий (их сводка – см. приложение).

Образующие группы G – это элементы, через произведение которых и обратных к ним можно выразить любой элемент группы. Нас будут интересовать только *конечно-порожденные* группы, т.е. группы, имеющие конечный набор образующих. По группе G с фиксированном набором образующих можно построить $\Gamma(G)$ – граф группы G. (Один из таких графов и даст нам искомый пример.) Пусть, например, G имеет две образующие – a и b. Вершинами этого графа будут элементы группы G, два элемента g_1 и g_2 соединены ребром $\iff$ выполнено одно из равенств: $g_1 = g_2 a^{\pm 1}$ или $g_1 = g_2 b^{\pm 1}$[2]

Простейший пример: a и b есть параллельные переносы на единичные векторы $\vec{e}_x$ и $\vec{e}_y$.Тогда G есть группа переносов на целочисленные векторы и $\Gamma(G)$ есть обычная целочисленная решетка. Вот более сложный пример.

Пример. *Свободная группа с двумя образующими* – это множество слов, состоящих из символов a, b, a^{-1}, b^{-1}. Слова умножаются простым приписыванием, с последующем сокращением букв с обратными буквами. Пусть, к примеру, $u = abab^{-1}ab, v = b^{-1}a^{-1}ba^2b$. Тогда $vu = b^{-1}a^{-1}ba^2babab^{-1}ab$, а $uv = aba^3b$ ($a^n = a \dots a$). Как видите, $uv \neq vu$. Пустое слово Λ, не содержащее символов, соответствует единице группы. $\Lambda = aa^{-1} = bb^{-1}$. Граф $\Gamma(G, a, b)$ показан на рисунке:

Упражнения. Расстояние между двумя вершинами графа – это минимальное число звеньев пути, их соединяющего. (Если такого пути нет, то расстояние считают бесконечным.)

а) Докажите, что выбор других образующих может изменить расстояния в $\Gamma(G)$ только в ограниченное число раз.

б) Докажите, что $\Gamma(G)$ всегда связен и однороден, т.е. любую его вершину можно перевести в любую другую подходящим автоморфизмом графа (автоморфизм – это изоморфизм на себя, уважающий ребра). Как устроен этот автоморфизм?

в) Если в определении графа группы соединять ребром вершины, отличающиеся только левым умножением на образующую (ранее мы соединили вершины, отличающиеся только правым умножением), то получится граф Γ', совпадающий с Γ, тогда и только тогда, когда G – коммутативна. Докажите, что и в общем случае графы Γ и Γ' изоморфны.

Элемент $g \in G$ группы G называется *периодическим*, если при некотором $n \geq 1$ $g^n = e$ - единице группы G, иначе g называется *непериодическим*. Группа G называется *периодической*, если все ее элементы периодические. Иначе, т.е. если в G имеется непериодический элемент x, то G называется *непериодической*.

[2] Иногда мы будем писать $\Gamma(G, \{a_i\})$, где $\{a_i\}$ – набор образующих.

Задача. Пусть G – непериодическая группа. Расположив три флажка по направлению степеней непериодического элемента x и действуя как в задаче … нарисуйте схему обхода ее графа с помощью трех флажков.

А вот можно ли обойти граф периодической группы? Если она конечна – то без сомнения. Однако имеет место следующий

Факт. (Теорема Бернсайда) *Существует бесконечная конечно-порожденная (даже 2-порожденная) периодическая группа G.*

Так вот, граф группы G – это то, что нам надо. Справедлива

Теорема. $\Gamma(G)$, *вне зависимости от выбора набора образующих G не обходится никаким конечным семейством Роботов и флажков. Более того, всегда можно указать конечную область в $\Gamma(G)$, которую заданное конечное семейство роботов и флажков никогда не покинет.*

Доказательство. Посмотрим сначала, как движется Робот по Γ один, без флажков. Поскольку наш граф в каждой своей вершине устроен одинаково, то Робот зациклится, а значит, его движения начнут периодически повторятся. Посмотрим, каково смещение за период. Из вершины $g \in G$ Робот идет в вершину $ga_i^{\pm 1}$, в зависимости от своего внутреннего состояния. Когда пройдет период и два его состояния повторятся, Робот придет в вершину $g \cdot h$, где

$$h = a_{i_1}^{\varepsilon_1} \cdots a_{i_n}^{\varepsilon_n},$$

$\varepsilon_i = \pm 1$, a_i и ε_i определяются внутреним состоянием Робота в соответствующий момент. Когда минуют два периода, Робот перейдет в вершину $g \cdot h^2$, а когда n — $g \cdot h^n$. Но раз G периодична, то при некотором n $h^n = e$ и, стало быть, Робот ходит по кругу! Он не может далеко уйти от того места, куда его поместили изначально! (А если у Робота есть предпериод, то его траектория суть "цикл с хвостом".) Итак, имеется функция F, такая, что автомат, имеющий только n внутренних состояний, не покинет области диаметром $F(n)$, вне зависимости от его начального положения.

Теперь посмотрим, помогут ли Роботу флажки выйти из аколдованного круга. Посмотрим сперва, как меняет дело один флажок. А никак! Ведь Робот не может далеко уйти от флажка и, стало быть, система Робот-Флажок может принимать только конечное число состояний и ее можно рассматривать как виртуальный конечный автомат! Можно (хотя и не обязательно) оценить число его внутренних состояний. Пусть k – число образующих группы G. Тогда вершин в графе Γ, находящихся на расстоянии не выше l от заданной не более чем $(2k)^l$. Поэтому число состояний виртуального автомата не более, чем $n \cdot (2k)^{F(n)}$ и система робот-флажок не покинет области диаметром $F(n \cdot (2k)^{F(n)})$.

Рассмотрим случай двух флажков. Если расстояние между ними ограничено, то проходит идея виртуального автомата с теми же рассуждениями. Иначе Робот должен далеко разнести флажки и, стало быть, далеко идти только с одним флажком, что, как мы уже доказали, разбирая случай одного флажка, невозможно.

Упражнение. Уточните приведенные рассуждения. Докажите, что система "Робот с двумя флажками"не покинет область диаметра

$$F\left(n \cdot F(n \cdot (2k)^{F(n))^3}\right)$$

Теперь общий случай, когда флажков много, скажем 2004. Сделав предположение индукции, мы можем считать, что система "Робот менее чем с 2004 флажками"не покинет ограниченной области размера $H(n)$ (n – число внутренних состояний Робота). А дальше рассуждать аналогично: либо расстояния между флажками неограниченно – тогда система Робот менее чем с 2004 флажками совершает неограниченное путешествие, что невозможно. Либо, если такое расстояние ограниченно то применим идею виртуального автомата.

Недаром герои сказок в заколдованном лесу хоят по кругу, мечтая найти прямой путь!

Упражнение. а) Получите оценки, аналогичные предыдущему упражнению.

б) Проведите те же рассуждения для общей системы из нескольких Роботов и флажков.

Мы расскажем теперь, как додуматься до того, чтобы взять граф бесконечной периодической группы в качестве непроходимого лабиринта? Ясно, что конструкцию графа группы, понятие периодической группы и формулировку теоремы Бернсайда надо было знать заранее. Для непроходимости лабиринта нужно, чтобы на флажках нельзя было бы организовать Машину Минского, а Робот ее организует, как мы это видели на примере задачи …, ставя флажки вдоль своей траектории. Поэтому у Робота не должно быть длинных путей, что можно обеспечить зацикливанием. Из эстетических соображений хочется, чтобы граф был бы одинаково устроеным во всех вершинах, т.е. однородным, а в качестве однородного графа естественно брать граф группы. Чтобы пути зациклились, группа должна быть периодической, а чтобы Робот ее не обошел - бесконечной.

Если рассуждение содержательно, пусть ошибочно, то его обычно можно исправить и получить результаты, правда не те, которые ожидались первоначально.

Вы, вероятно, заметили одну странность в кодировке Машины Минского: когда нам требовалось запомнить несколько чисел $a_1, \ldots, a_k$, мы рассматривали число $2^{a_1} 3^{a_2} \ldots p_k^{a_k}$ (где $2, 3, \ldots, p_k$ – последолвательные простые числа). Операция возведения в квадрат тоже производилась как-то странно: мы рассматривали переход $2^n \to 2^n$. Встает вопрос: а нельзя ли все это делать естественнее, например написать программу для непосредственного перехода от n к n^2?

Оказывается, такие переходы невозможны. Чтобы это доказать, нам понадобится то самое рассуждение с периодическим поведением Робота на четвертинке плоскости, которое в тот раз оказалось ошибочным, теперь же мы его подправим.

Приглядимся к нему повнимательнее. Пусть два внутренних состояния Робота совпали на нижнем крае, и, кроме того, соответствующие координаты x_1 и x_2 оказлись сравнимыми по модулю N (которое делится на все, что нужно). В этом случае приход к другой стенке и взаимодействие с ней будут одинаковыми. Правда, соответствующие координаты y_1 и y_2 окажутся, вообще говоря, по модулю N несравнимыми и дальнейшее поведение Робота может измениться. Однако, если x_1 и x_2 будут сравнимы по модулю N^2, то y_1 и y_2 будут сравнимы по модулю N.

Покажем это. Из леммы ... следует, что $y = a/b \cdot x + c$, где числа a, b, c не превосходят по модулю числа внутренних состояний Робота. Поэтому $y_1 - y_2 = a/b \cdot (x_1 - x_2)$. По выбору N число b делит N.

В этом случае повторится и следующее взаимодействие со стенкой. Точно так же, если x_1 и x_2 сравнимы по модулю N^3, то $y_1 \equiv y_2 (mod N^2)$ и $x'_1 \equiv x'_1 (mod N)$ и уже взаимодействие со стенкой в точках y'_1 и y'_2 будет одинаковым. Аналогичным образом, если $x_1 \equiv x_2 (mod N^{2k})$, то $x_1^{(i)} \equiv x_2^{(i)} (mod N^{2(n-i)}$ и взаимодействие со стенками, а, значит, и поведение Робота в точках $x_1, x'_1, \ldots, x_1^{(n)}$и $x_2, x'_2, \ldots, x_2^{(n)}$ соответственно, будет одинаковым.

Задача. Найдите ошибку в этих рассуждениях.

Ошибка в том, что мы не учли эффект угла. Взаимодействие Робота со стенкой может зависеть и от того, встретит ли он по ходу дела еще и другую стенку. Однако если Робот все время далек от угла (т.е. все $x_i^{(j)}$ и $y_i^{(j)}$ больше удвоенного числа его внутренних состояний), то такого эффекта не возникает.

Таким образом, поведение Робота для начальных точек x и $x + aN^{2k}$ при любых x и a до осуществления $2k$-го удара о стенку будет одинаковым, за исключением только что сделанной оговорки.

Теперь мы можем доказать, что программы перехода от n к n^2, универсальной по n для Робота с ограниченным числом состояний не существует.

В самом деле: предположим, что такая программа существует. Пусть l – число внутренних состояний Робота. Прежде всего заметим, что приближение Робота к углу будет происходить только для конечного числа начальных позиций n, так что весь путь Робота, начинающийся с достаточного большого n будет лежать достаточно далеко от угла и соответствующих эффектов не будет. В противном случае из-за конечности числа полей, близких к вершине угла, на какое-то из них Робот прийдет в одном и том же состоянии, но из разных начальных положений n_1 и n_2. В этом случае его дальнейшее поведение, а, значит, и финальное положение будет одинаковым. Но $n_1^2 \neq n_2^2$! Противоречие.

Итак, пусть n – достаточно большое число и пусть в процессе перехода от n к n^2 произошло k ударов о стенки. Возьмем $n_1 = n, n_2 = n_1 + a \cdot N^k$. В результате из новой позиции n_2. Робот произведет те же k ударов о стенку и точно так же остановится. И все режимы его движения между ударами будут в точности такими же!

В первом случае Робот совершил переход $n \to n^2$. Из точки $n + aN^k$ Робот должен перейти в точку $n^2 + 2naN^k + a^2N^{2k}$ при любом a за те же k ударов!

Почему это невозможно? Потому, что расстояние выросло в $n + aN^k$ раз, а за один удар оно может измениться не более чем в l раз (почему?) и при $a > l^k$ такое невозможно. Как видите, рассуждение может иметь ценность само по себе, хотя в одних случаях оно может оказаться ошибочным, в других – нет.

Задачи.

1. Докажите, что не существует программы перехода $n \to n!$, универсальной по n.

2. Докажите, что не существует программы, позволяющей Роботу, стартующему с начального положения n, приходить на исходную позицию, если n – степень двойки, а в противном случае – приходить в вершину угла.

3. Пусть N – то самое число, делящее всевозможные периоды Робота. Пусть Робот осуществил переход $X \to Y$, сделав k ударов о края, и ни разу не приблизился к вершине угла в том смысле, который обсуждался выше. Докажите, что из начального положения $X + l \cdot N^k$ он осуществит переход в положение $Y + l \cdot \lambda \cdot N^k$, где λ – рациональное число, не зависящее от l.

Другой близкий сюжет связан с *распознаванием.* Пусть Робот путешествует по прямоугольнику $n \times k$. Его цель – проверить, удовлетворяют или нет n и k данному соотношению. Например, идя под углом в 45^o Робот легко проверит справедливость равенства $k = n$. Ввыходя же из левого верхнего угла и идя 2 раза вправо и 1 раз вниз он проверит справедливость или несправедривость равенства $k = n/2$.

С помощью Леммы... решается следующее

Упражнение. а) Пусть $\alpha, \beta \in \mathbf{Q}$. Докажите, что существует Робот, распознающий равенство $k = \alpha \cdot n + \beta$. Число его состояний зависит от α и β, но не зависит от n и k.

б) Аналогичным образом, докажите, что для заданной системы $\mathfrak{R}$ линейных уравнений и неравенств с рациональными коэффициентами от двух переменных, распознается ее выполнимость на k и n.

в) Если $k \leq n$, то справедливость равенства $k = 2^l$ тоже распознается.

Итак, "линейные вещи"распознаются. А вот распознаются ли другие, более сложные равенства, например, равенство $k = n^2$? Длина программы не должна зависеть от n и k.

Оказывается, даже такое простое равенство не распознается и идея доказательства близка к идее доказательства невозможности перехода $n \to n^2$. Дадим **набросок доказательства**.

Посмотрим, как движется Робот, имеющий l внутренних состояний вдоль полосы шириной n. Его состояния периодически повторяются, причем период не превосходит $n \cdot l$, если же последовательно выписать его состояния только на данном расстоянии от края, то период смены таких состояний не больше l. За каждое соударение он сдвигается вдоль полосы на $\alpha_i n + \beta_i$, а, стало быть, раз когда его состояния на данном расстоянии от края повторятся, произойдет не более $2l$ ударов о края и Робот сдвинется на число указанного вида. При этом величины α_i и β_i рациональны и для них есть только конечное число возможных значений в зависимости от l. Рассмотрим всевозможные величины сдвигов Робота менее чем с l состояниями в полосе ширины n соответствующие ближайшим повторениям его состояний. Пусть $k_1 \equiv k_2$ по всем таким модулям. Если $l \cdot n < k_1 < k_2$, то эффекта предпериода не возникает и прямоугольники $n \times k_1$ и $n \times k_2$ не различаются Роботом. Ясно, что при больших n $n \cdot l < n^2$ и всегда найдется число k, большее n^2 сравнимое с n^2 по данному набору модулей, тогда прямоугольники $n \times k$ и $n \times k^2$ неразличимы. Точно так же доказывается нераспознаваемость равенств $k = n^3$ и $k = n!$

Упражнение. а) Оформите набросок в доказательство. Говоря точнее, при $k > 2ln$ существует такое t, зависящее от n и l, что при $m \in \mathbf{N}$ прямоугольники вида $n \times k$ и $n \times (k+mt)$ не различаются Роботом, в куда бы его не поместили изначально.

б) Рассмотрим движение Робота в параллепипеде $n \times m \times k$. Пусть $k > 2lmn$. Тогда при некотором t зависящим от l, m, n для любого i, если $k' = k + i \cdot t$, то параллепипеды $n \times m \times k$ и $n \times m \times k'$ неразличимы.

в). Исследуйте другие случаи возможности и невозможности распознать.