

image not found or type unknown



Ханойские башни - это игра, в которой используются три штыря и набор дисков. Все диски различаются диаметром и нанизываются на штыри через отверстие в центре каждого диска. Первоначально все диски находятся на левом штыре.

Цель игры состоит в том, чтобы переместить все диски на центральный штырь. Правый штырь можно использовать как «запасной» для временного размещения дисков. При каждом перемещении диска с одного штыря на другой должны соблюдаться два ограничения: перемещать можно только самый верхний диск на штыре, и, кроме того, нельзя ставить диск на другой диск меньшего размера.

В данном эссе будет рассмотрен способ решения поставленной задачи «Ханойские башни», связь этой задачи с теорией графов. Так же будут рассмотрены рекурсия, на примере данной задачи и код Грея.

1. История задачи «Ханойские башни»

Эту известную игру придумал французский математик Эдуард Люка, в 1883 году её продавали как забавную игрушку. Первоначально она называлась «Профессор Клаус из Колледжа Ли-Су-Стьян» но вскоре обнаружилось, что таинственный профессор из несуществующего колледжа - не более чем анаграмма фамилии изобретателя игры - профессора Люка из колледжа Сен-Луи.

Но история этой задачи уходит довольно глубже. Легенда гласит, что в Великом храме города Бенарас, под собором, отмечающим середину мира, находится бронзовый диск, на котором укреплены 3 алмазных стержня, высотой в один локоть и толщиной с пчелу. Давным-давно, в самом начале времён, монахи этого монастыря провинились перед богом Брахмой. Разгневанный, Брахма воздвиг три высоких стержня и на один из них возложил 64 диска. Брахма поместил на один из стержней 64 диска из чистого золота, причем так, что каждый меньший диск лежит на большем.

Как только все 64 диска будут переложены со стержня, на который Брахма сложил их при создании мира, на другой стержень, башня вместе с храмом обратятся в пыль и под громовые раскаты погибнет мир.

Количество переключений в зависимости от количества колец вычисляется по формуле $2^n - 1$.

Число перемещений дисков, которые должны совершить монахи, равно **18 446 744 073 709 551 615**. Если бы монахи, работая день и ночь, делали каждую секунду одно перемещение диска, их работа продолжалась бы **584 миллиарда лет**.

В информатике задачи, основанные на легенде о Ханойской башне, часто рассматривают в качестве примера использования рекурсивных алгоритмов и преобразования их к не рекурсивным.

2. Суть задачи

Ханойская башня является одной из популярных головоломок XIX века. Даны три стержня из N дисков разного диаметра, которые надеты на стержень (1) в порядке убывания диаметра (Рис. 1).

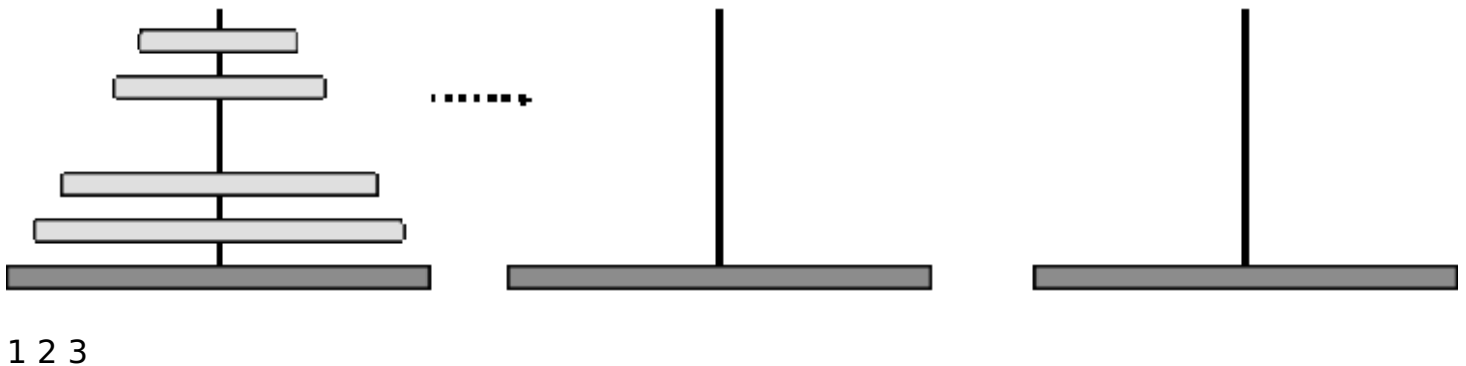


Рис. 1 (стержни и диски)

Надо переместить N дисков за наименьшее число шагов на стержень (3), так чтобы они остались в таком же порядке. При этом требуется соблюдать правила:

- На каждом шаге ровно один диск перемещается с одного диска на другой;
- Диск большего диаметра нельзя помещать на диск меньшего диаметра;
- Стержень (2) можно использовать как промежуточный.

3. Построение модели

Математической моделью данной задачи является рекуррентное соотношение.

Рекуррентное соотношение - это соотношение, которое выражает значение функции с помощью других значений, вычисленных для меньших аргументов.

Исходя из данного определения, следует, что для каждой рекуррентной функции нужно задавать хотя бы одно значение.

4. Решение с помощью рекурсии

Многим из тех людей, которые играют в эту игру, практически никогда не удастся обнаружить весьма простую стратегию, позволяющую успешно играть в Ханойские башни с тремя штырями и N дисками. Чтобы не утомлять вас поисками решения этой задачи, мы откроем его:

- Граничное условие выполняется в случае, когда на исходном (левом) штыре нет дисков.
- Переместить $N-1$ дисков с исходного штыря на запасной (правый) штырь, используя итоговый штырь как запасной; отметим, что это перемещение осуществляется рекурсивно.
- Переместить один диск с исходного штыря на итоговый штырь. В этом месте наша программа будет выдавать сообщение об этом перемещении.
- Наконец, переместить $N-1$ дисков с запасного на итоговый, используя исходный штырь в качестве запасного.

Начнем с самого маленького кольца и переложим его на любую отметку. В дальнейшем это кольцо нужно перемещать в том же направлении, что и при первом переключении. Затем произведем единственно возможное перемещение оставшихся колец, после чего снова переложим самое маленькое кольцо и т. д. (Интересно заметить, что, перенумеровав «кольца» по порядку, мы добьемся неожиданного эффекта: четные кольца будут перемещаться из одной вершины треугольника в другую в одном направлении, а нечетные - в противоположном направлении.)

Задача о ханойских башнях - это классический пример применения рекурсии для описания эффективного алгоритма.

В программировании рекурсия - вызов функции (процедуры) из неё же самой, непосредственно (простая рекурсия) или через другие функции (сложная или косвенная рекурсия), например, функция A вызывает функцию B , а функция B - функцию A . Количество вложенных вызовов функции или процедуры называется глубиной рекурсии.

Преимущество рекурсивного определения объекта заключается в том, что такое конечное определение теоретически способно описывать бесконечно большое число объектов. С помощью рекурсивной программы же возможно описать бесконечное вычисление, причём без явных повторений частей программы.

Займемся теперь непосредственно методом, реализующим алгоритм и перекладывающим кольца в соответствии с правилами игры. Написать нерекурсивный вариант ханойских башен совсем не просто. Можно, конечно, написать цикл, завершающийся по достижению требуемой конфигурации, на каждом шаге которого выполняется очередной ход. Но даже первый ход не тривиален. Поскольку фиксирован столб, где должны быть собраны кольца, то неясно, куда нужно переложить первое кольцо - на второй или третий столб?

Рекурсивный вариант решения задачи прозрачен, хотя и напоминает некоторый род фокуса, что характерно для рекурсивного стиля мышления. Базис рекурсии прост. Для перекладывания одного кольца задумываться о решении не нужно - оно делается в один ход. Если есть базисное решение, то оставшаяся часть также очевидна. Нужно применить рекурсивно алгоритм, переложив $n-1$ кольцо с первой пирамиды на третью пирамиду. Затем сделать очевидный ход, переложив последнее самое большое кольцо с первой пирамиды на вторую. Затем снова применить рекурсию, переложив $n-1$ кольцо с третьей пирамиды на вторую пирамиду. Задача решена.

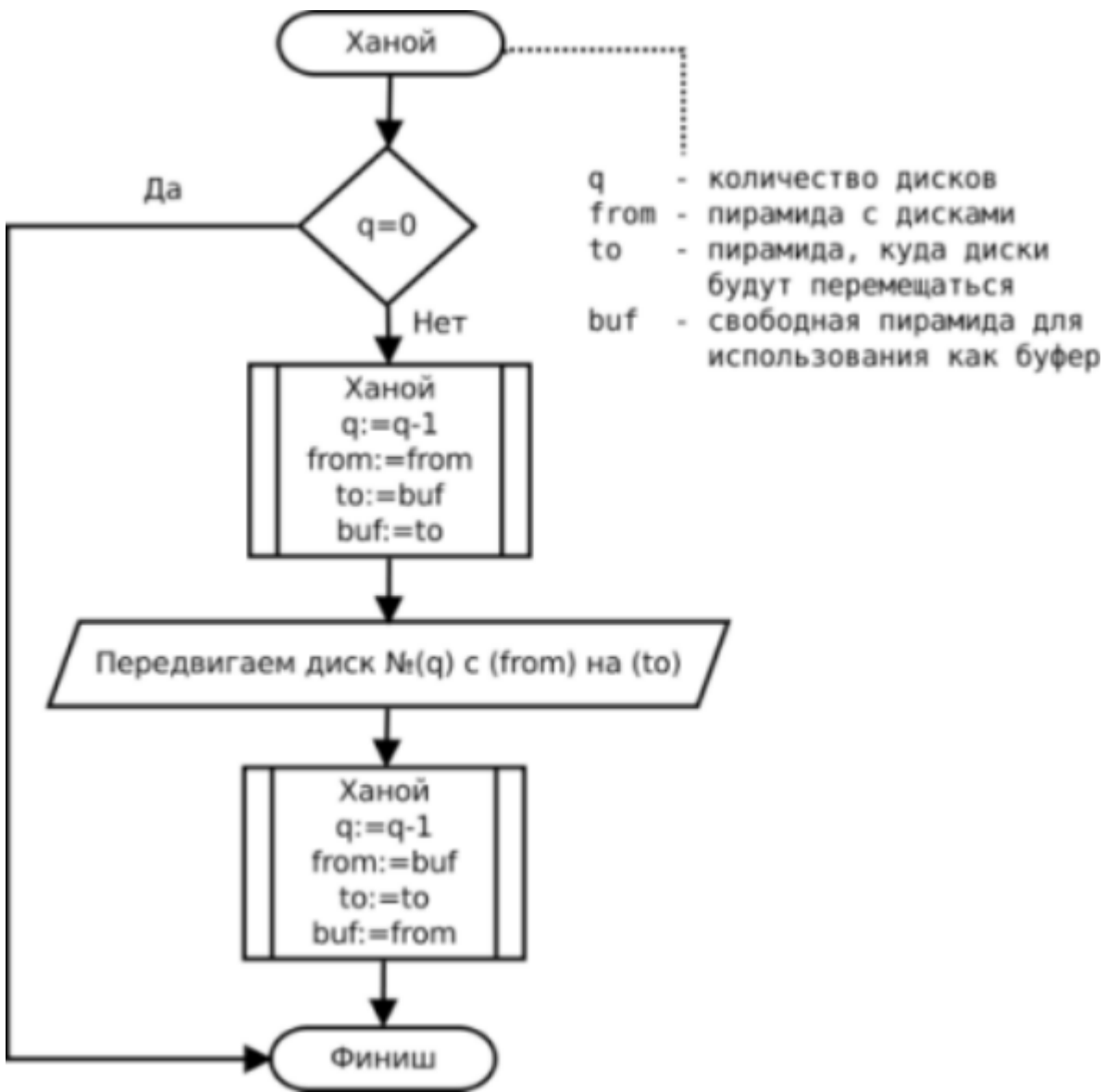


Рис. 2 (блок-схема алгоритма)

5. Сложность и затраты времени

Проведем анализ временных затрат для ханойских башен (и всех задач, сводящихся к решению двух подзадач размерности $n-1$). Подсчитаем требуемое число ходов $T(n)$. С учетом структуры решения:

$$2T^{(n-1)}$$

$$T(n) = \quad +1$$

Простое доказательство по индукции дает:

$$T(n) =$$

2^n 2^n 2^n

$$-1 + -2 + \dots + 2 + 1 = -1$$

Можно показать, что последовательность ходов, реализуемая рекурсивным алгоритмом, является оптимальной, так что никакой другой алгоритм не может решить задачу за меньшее число ходов.

Алгоритм решения задачи о Ханойских башнях является конечным, так как все используемые циклы выполняются конечное число раз.

Сложность - количественная характеристика алгоритма, которая говорит о том, сколько времени он работает (временная сложность), либо о том, какой объем памяти он занимает (емкостная сложность). На практике сложность рассматривают как временную сложность.

Из определения сложности следует, что она зависит от размерности входных данных или, как говорят, от длины входа. В задаче о Ханойских башнях входными данными является число дисков.

Рассчитаем порядок временной сложности в соответствии с пошаговым алгоритмом.

Временная сложность процедуры будет зависеть от количества переносов, которое равно $2^n - 1$, значит $O(2^n - 1)$.

6. Связь задачи «Ханойские башни» с теорией графов

Если внимательно вдуматься в задачу, то можно предположить, что все диски можно переложить на стержень (2) с помощью стержня (3) и наоборот. Т.е. у нас появляется выбор. Следовательно есть несколько способов решить эту задачу.

Еще одно замечательное следствие состоит в том, что в процессе перекладывания колец встретятся все допустимые расположения колец на трёх стержнях, так сказать, «все позиции». Почему это можно утверждать?

Потому, что общее число позиций равно 2^n (чтобы задать позицию, мы для каждого из n колец должны выбрать тот из трёх стержней, на котором оно в этой позиции находится - а порядок колец на стержне жёстко задан их размерами), и по той простой причине, что никакая позиция не встречается в процессе перекладывания дважды (иначе процесс можно было бы сократить). А поскольку мы вынуждены делать именно $(2^n - 1)$ ходов, каждый раз получая неповторяющиеся позиции, то

все позиции будут получены.

В задаче было показано, что путь от исходного (все кольца собраны на левом стержне) до конечного (все кольца собраны на правом из трех стержней) состояния проходит абсолютно через все допустимые позиции. Осталось только осмыслить, как именно это происходит. Для этого надо изобразить все возможные позиции игры в виде графа:

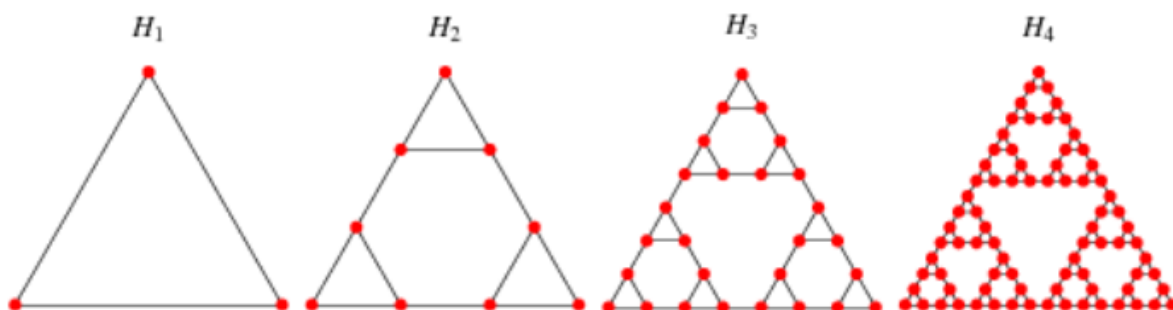


Рис. 3 (граф, демонстрирующий выбор позиций дисков)

На рис. 3 показаны графы H_1 - H_4 башен с числом колец от 1 до 4. Ходы по сторонам самых маленьких треугольников соответствуют переносу самого маленького кольца, ходы между такими треугольниками (но внутри треугольника следующего размера) - переносу следующего кольца, и т. д. Это в стандартной головоломке, когда все переносы разрешены. А что получается, если между стержнями А и С ничего переносить нельзя?

Картинка с изображением графа, который получается из H_3 , приведена на рис. 4:

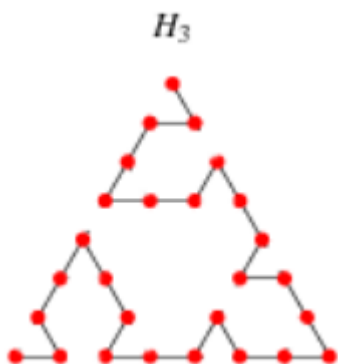


Рис. 4 (ограниченный условием граф)

Попробуем понять, почему граф «обрезается» именно так, как показано на этой картинке. Для этого «укрупним» изображения точек на исходном графе и поместим

на них информацию о том, какому расположению колец соответствует данная точка (рис. 5):

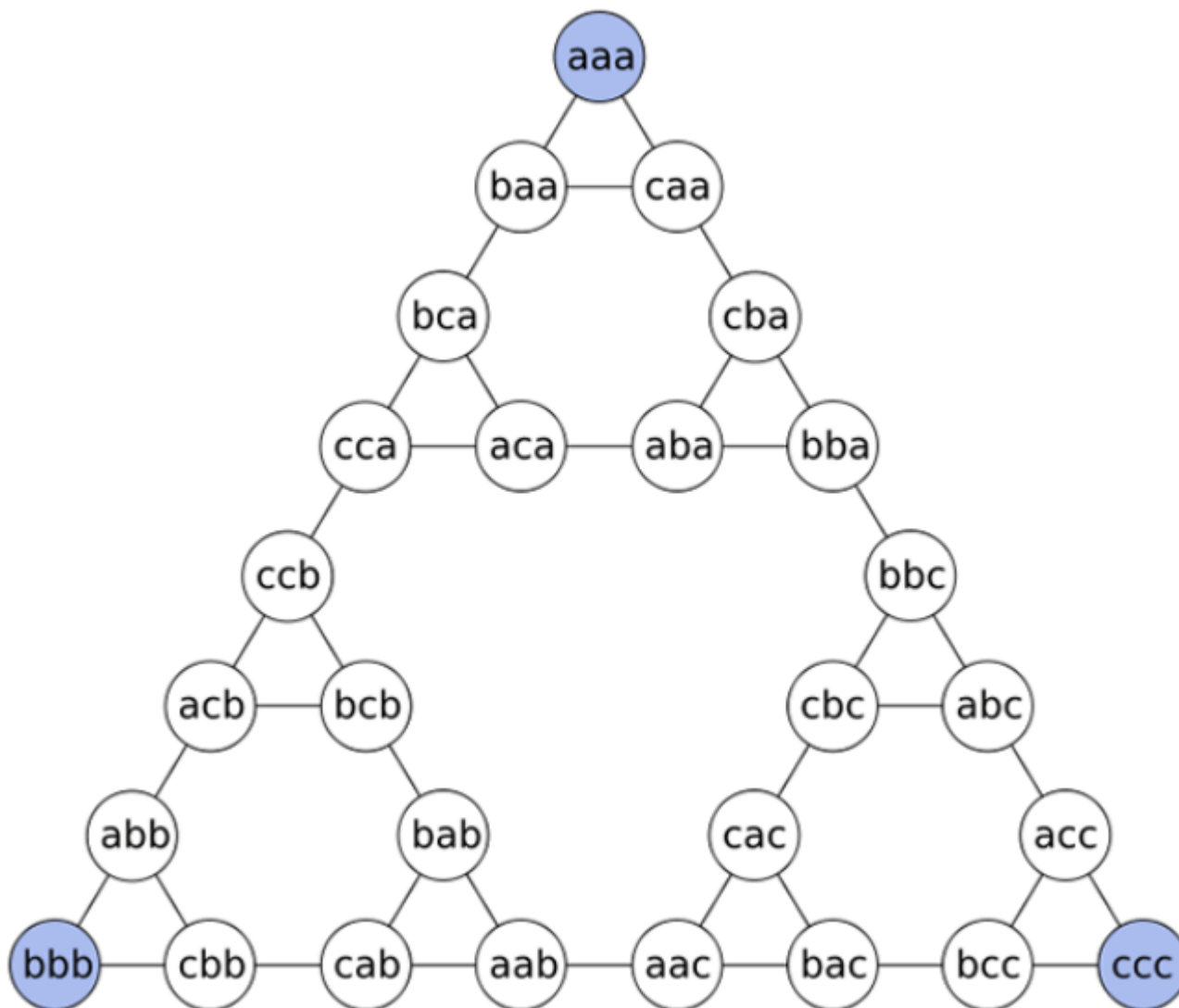


Рис. 5 (граф с координатами дисков)

Здесь «cbb», например, означает, что самое маленькое кольцо находится на стержне С, а оба остальных - на стержне В. При этом можно либо перенести маленькое кольцо (в стандартном графе - на любой из стержней А или В, то есть получить abb или bbb), либо перенести следующее кольцо на свободный стержень (то есть получить cab). В новой версии переносы с А на С и с С на А запрещены, так что ровно один из этих трех ходов оказывается невозможным, - а это означает, что каждая позиция, кроме первой и последней, имеет ровно две соседних. Тем самым весь путь по графу оказывается одной длинной ломаной линией без всяких ответвлений.

Интересно, что построенный нами путь по графу (точнее, его буквенная запись вида **aaa - baa -... - ccc**) встречается в очень многих серьезных математических (и не только математических) исследованиях и носит специальное название - код Грея

<http://ru.wikipedia.org/wiki/%D0%9A%D0%BE%D0%B4_%D0%93%D1%80%D0%B5%D1%8F>

Главная особенность кодов Грея состоит в том, что соседние значения в последовательности «кодовых слов» отличаются только в одном разряде.

7. Применение кода Грея для решения

башня рекурсия граф головоломка

Коды Грея применяются в решении задачи о Ханойских башнях. Пусть N - количество дисков. Начнём с кода Грея длины N , состоящего из одних нулей (то есть $G(0)$), и будем двигаться по кодам Грея (от $G(i)$ переходить к $G(i+1)$). Поставим в соответствие каждому l -ому биту текущего кода Грея l -ый диск (причём самому младшему биту соответствует наименьший по размеру диск, а самому старшему биту - наибольший).

Поскольку на каждом шаге изменяется ровно один бит, то мы можем понимать изменение бита l как перемещение l -го диска. Заметим, что для всех дисков, кроме наименьшего, на каждом шаге имеется ровно один вариант хода (за исключением стартовой и финальной позиций). Для наименьшего диска всегда имеется два варианта хода, однако имеется стратегия выбора хода, всегда приводящая к ответу: если N нечётно, то последовательность перемещений наименьшего диска имеет вид $f \rightarrow t \rightarrow r \rightarrow f \rightarrow t \rightarrow r \rightarrow \dots$ (где f -стартовый стержень, t -финальный стержень, r -оставшийся стержень), а если N чётно, то $f \rightarrow r \rightarrow t \rightarrow f \rightarrow r \rightarrow t \rightarrow \dots$

8. Различные задачи с измененным условием

Ханойские башни - старая задача и она уже давно решена, поэтому сейчас существует уже много задач, условие которых было изменено или доработано для усложнения.

В данном эссе были рассмотрены некоторые из них:

Постановлением ЮНЕСКО оригинал Ханойской башни был подвергнут реставрации. В связи с этим во время пользования головоломкой нельзя было переключать кольца с первого стержня сразу на третий и наоборот.

Решите головоломку (переложите все кольца с первого стержня на третий) с учетом этих ограничений. Вам не нужно находить минимальное решение, но количество совершенных перемещений не должно быть больше 200000, при условии, что количество дисков не превосходит 10. Каждое перемещение задается тремя числами: номер кольца, исходный стержень, конечный стержень (приложение 3).

На дорогах Ханоя было введено одностороннее круговое движение, поэтому теперь диск со стержня 1 можно перекладывать только на стержень 2, со стержня 2 на 3, а со стержня 3 на 1.

Решите головоломку с учетом этих ограничений. Вам не нужно находить минимальное решение, но количество совершенных перемещений не должно быть больше 200000, при условии, что количество дисков не превосходит 10.

Заключение

Во время выполнения эссе были изучены следующие вопросы:

- 1) Алгоритм рекурсии;
- 2) Код Грея;
- 3) Связь задачи «Ханойские Башни» с теорией графов;
- 4) Некоторые методы работы с языком программирования C# ;

Была составлена программа для наглядного представления работы рекурсии на примере задачи о Ханойских башнях на C# в VisualStudio 2010. После проведенных тестов, был сделан вывод, что программа работает корректно, следовательно, поставленная задача выполнена.

Список используемой литературы

1. С. М. Окулов, А. В. Лялин «Ханойские Башни» 2008г. 248 стр.
2. Сайт «Wikipedia» http://ru.wikipedia.org/wiki/Ханойская_башня
3. Сайт «элементы» <http://elementy.ru/problems/441>
- . Сайт «C# Programming» <http://www.c-help.net/142.html>

5. Сайт informatics.mccme <http://informatics.mccme.ru/moodle/mod/statements/view3.php?id=2550&chapterid=3050>

Нормативные ссылки

В данной пояснительной записке использованы ссылки на следующие стандарты:

ГОСТ Р 1.5-2004. Стандарты национальные РФ. Правила построения, изложения, оформления и обозначения.

ГОСТ 2.301-68 ЕСКД. Форматы.

ГОСТ Р 7.0.5-2008 СИБИД. Библиографическая ссылка. Общие требования и правила составления.

ГОСТ 7.12-93 СИБИД. Библиографическая запись. Сокращения слов на русском языке. Общие требования и правила.

ГОСТ 7.82-2001 СИБИД. Библиографическая запись. Библиографическое описание электронных ресурсов. Общие требования и правила составления.