

Алгоритмы. Перебор и рекурсия.

Информатика
10-11 классы

29 ноября 2011 г.

Что такое алгоритм?

- Алгоритм — это конечный набор правил, который определяет последовательность операций для решения конкретного множества задач и обладает пятью важными чертами: конечность, определённость, ввод, вывод, эффективность (Д. Кнут).
- Основные свойства алгоритма:
 - ❶ Дискретность — последовательное выполнение простых шагов.
 - ❷ Детерминированность — определённость — одинаковые исходные данные дают одинаковый результат.
 - ❸ Понятность.
 - ❹ Конечность — количество шагов должно быть конечно (может быть и неявно задано).
 - ❺ Универсальность — алгоритм должен работать для множества исходных данных.

Виды алгоритмов

- 1 Динамическое программирование (было).
- 2 Метод перебора.
- 3 Рекурсия (повторение через себя). *Рекурсия — это рекурсия.*
- 4 Жадные алгоритмы (хватай сначала самое большое).
- 5 Разделяй и властвуй (привет, принцип 3 ветвей власти).
- 6 Метод двоичного поиска.
- 7 Много–много других.

Метод перебора

- Идея метода перебора очень проста — переберём все варианты и выберем только те, которые нам нужны.
- Рассмотрим простую задачу: вывести на экран простые числа $\leq N$.
- Есть красивые решения типа решета Эратосфена. Но мы пойдём “в лоб”.

Listing 1: Простые числа

```
def is_prime?(n)
  return false if ((n == 1) || (n == 0))
  for i in 2..(n-1)
    return false if (n%i == 0)
  end
  true
end

100.times {|i| puts i if is_prime?(i) }
```

Метод перебора: НОД

- Рассмотрим задачу: найти $\text{НОД}(a, b)$.
- Пример: $\text{НОД}(15, 12) = 3$
- Решаем методом перебора.
- Переберём все возможные делители этих двух чисел.
- Для этого пройдемся циклом от 2 до, например, a .
- Если оба числа делятся на пробегаемое число, значит, пробегаемое число — делитель. Запоминаем его.
- Если в результате работы программы мы находим делитель, который больше запомненного ранее, то стираем старый и записываем новый.
- Изначально зададим $\text{НОД} = 1$ (на 1 все числа делятся).
- НОД по-английски — gcd.

Метод перебора: НОД

Listing 2: НОД(a

```
def gcd(a,b)
    max = 1
    for i in 2..a
        max = i if ((a%i==0) && (b%i==0))
    end
    max
end

a = 126
b = 486

puts gcd(a,b)
```

Задачи на метод перебора

- Задача 1. Дано некоторое число a . Вывести на экран все числа, которые взаимно просты с a и не превышают N . N задаётся также в начале программы.
- Пример: $a = 3$, $N = 10$. Программа должна вывести следующие числа: 2, 4, 5, 7, 8, 10.
- Задача 2. Вывести на экран все пары взаимно простых чисел, каждое из которых не превышает N . N задаётся в начале программы.
- Пример: $N = 5$. Программа должна вывести: (2,3), (2,5), (3,4), (3,5), (4,5).

Рекурсия

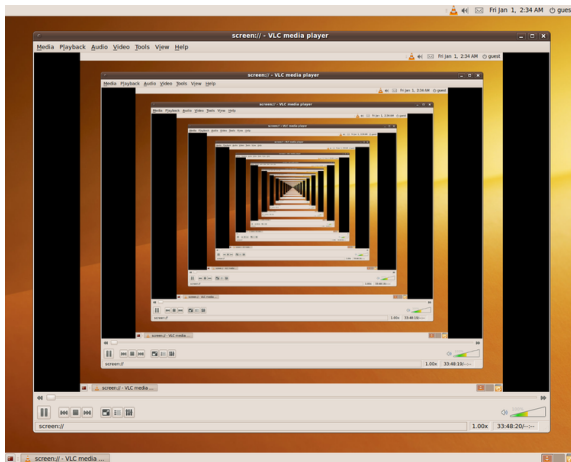


Рис.: Источник: <http://ru.wikipedia.org>

Рекурсия

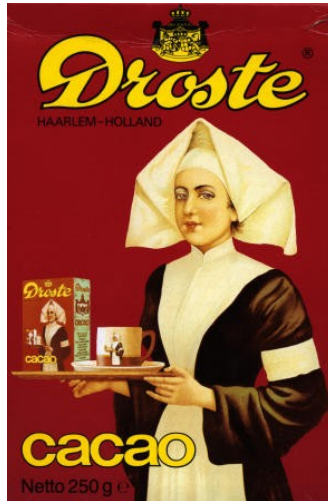


Рис.: Источник: <http://ru.wikipedia.org>

Рекурсия

- Рекурсия — это рекурсия.
- Рекурсия — это когда функция использует сама себя.
- Пример рекурсии — числа Фибоначчи. Для того, чтобы посчитать 10 число Фибоначчи, нужно знать 8 и 9. А чтобы узнать их, нужно знать 2 их предыдущих и так далее.
- У рекурсии всегда есть две составляющие:
 - 1 База — до какого момента спускаемся. В случае чисел Фибоначчи база — это нулевое и первое числа, которые равны 1.
 - 2 Рекуррентная формула — формула, которая говорит, как получить что-то из предыдущего. Для чисел Фибоначчи это: число Фибоначчи равно сумме двух предыдущих ($\varphi_n = \varphi_{n-1} + \varphi_{n-2}$)
- Рекурсия может быть медленной.

Числа Фибоначчи: рекурсия

- Напишем функцию для вычисления n -ного числа Фибоначчи через рекурсию.
- (!) Сравните по времени, сколько вычисляется 50-е число Фибоначчи рекурсией и обычным циклом.

Listing 3: Рекурсия для чисел Фибоначчи

```
def fibonacci(n)
    return 1 if ((n==0) || (n==1))
    fibonacci(n-1)+fibonacci(n-2)
end

puts fibonacci(10)
```

Легенда о Ханойских башнях

- 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.

Задача о ханойских башнях

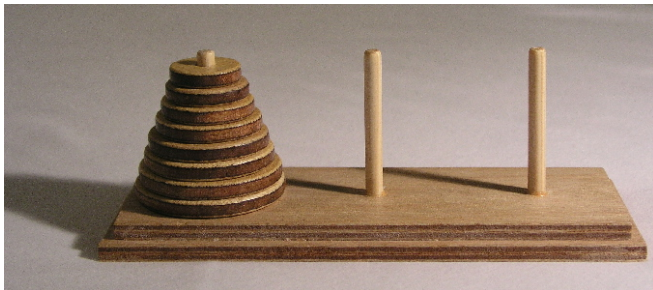


Рис.: Источник: <http://ru.wikipedia.org>

Задача о ханойских башнях

- Пример простого решения рекурсией **без понимания** работы алгоритма — задача о ханойских башнях.
- Даны три стержня, на один из которых нанизаны несколько колец, причем кольца отличаются размером и лежат меньшее на большем. Задача состоит в том, чтобы перенести пирамиду из восьми колец за наименьшее число ходов. За один раз разрешается переносить только одно кольцо, причём нельзя класть большее кольцо на меньшее.
- Пример алгоритма для 3 колец: с 1 на 3, с 1 на 2, с 3 на 2, с 1 на 3, с 2 на 1, с 2 на 3, с 1 на 3.
- Задача 1. Написать программу с использованием рекурсии, которая выводит на экран алгоритм решения задачи для заданного количества колец.
- Задача 2. Посчитать и объяснить, сколько минимально потребуется монахам времени, чтобы выполнить условие Брахмы. Предположим, что на 1 перекладывание монахи тратят 1 минуту.

References

- Все презентации доступны на <http://school.smirik.ru!>
- Вопросы, предложения, д/з: smirik@gmail.com