Вероятностные вычисления

Что такое случайность

Событие считается случайным, если оно происходит непредсказуемо.

Примеры:

- Подбрасывание монеты
- Выпадение игральной кости
- Шарик в рулетке

Цель

- Для решения некоторых труднорешаемых задач можно успешно применять случайность

Сравнение классических и вероятностных алгоритмов

Что лучше вероятностный алгоритм с очень маленькой ошибкой или классический (детерминированный), но работающий в тысячи раз дольше.

Классический алгоритм работает всегда правильно, но за время работы может дать сбой железо, на котором он выполняется.

Вероятность ошибки

- Что такое алгоритм с вероятностью ошибки 1/10000

- Если мы запустим алгоритм 10000 раз — получим правильный результат 9999 раз и ошибку всего один раз

Вероятностный алгоритм

- Детерминированный алгоритм при заданных входных данных выполняется однозначно.
- Вероятностный алгоритм при заданных входных данных может производить вычисления многими разными способами, один из которых выбирается случайным образом и выполняется

Первый способ выполнения вероятностного алгоритма

- Программа работает детерминировано за исключением нескольких операторов, в которых подбрасывается монета.
- Оператор: Flip a coin. If «орёл» then goto i else goto j.
- Подбросили монету, если «орел» перешли на строку i, если «решка» перешли на строку j

Второй способ выполнения вероятностного алгоритма

- Вероятностный алгоритм в начале своей работы выбирает один из нескольких вариантов выполнения
- Каждый из вариантов — детерминированный алгоритм. Для каждого примера задачи программа снова случайно выбирает новый алгоритм.

Пример вероятностного алгоритма.

- Задача: есть два компьютера R_1 и R_2

На R_1 - $x = x_1....x_n$, на R_2 - $y = y_1...y_n$. Время от времени надо сравнивать x и y.

- Детерминированный алгоритм требует передачи n бит.
- Вероятностному алгоритму достаточно 4log(n)

То есть для $n = 10^{16}$ — всего 256 бит

Вероятностный алгоритм WITNESS для равенства

- Шаг 1: R_1 выбирает простое число $p < n^2$
- Шаг 2: R_1 вычисляет $s = x \bmod p$ и отправляет s и p компьютеру R_2
- Шаг 3: R_2 получает s и p, вычисляет

$q = y \bmod p$

Если $q = s$, то R_2 выводит результат «равны»

Если $q \mathrel{!}= s$, то R_2 выводит результат «не равны»

Тестирование алгоритма WITNESS на входных данных

- Пример 1: x = 15(01111), y = 22(10110), n = 5

Множество простых чисел {2, 3, 5, 7, 11, 13, 17, 19, 23}

Пусть p = 5 - «x и y не равны»

Если p = 7 - «x и y равны»

Упражнения

Упр.1: Существует ли ещё простое число кроме 7 для предыдущей задачи, чтобы был дан неверный ответ для $x = 01111$ и $y = 10110$

Упр.2: Пусть $x = y = 100110$, существует ли простое число $p < 36$, что алгоритм даст неверный ответ.

Упр.3: Пусть $x = 10011011$ и $y = 010101016$ найти сколько простых чисел $p<64$ дадут правильный ответ и сколько неправильный

Оценка вероятности ошибок алгоритма

$Erorr_{WITNESS\ (x,\ y)}$ =

(число плохих простых чисел для (x,y)) / (количество всех простых чисел меньше n^2)

Высоконадежные вычисления

Допустим у нас есть шесть возможных простых чисел для какой-либо пары (x, y), и среди них одно не является свидетелем.

Вероятность ошибиться 1/6 = 0,17

Если повторить эксперимент пять раз, то мы ошибемся только, если все пять раз выберем не свидетеля, вероятность этого события

(1/6)*(1/6)*(1/6)*(1/6)*(1/6) = $1/6^5$ = 0,00013