

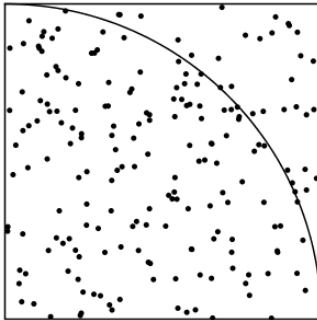
# Задача 1. Вычисление приближенного значения числа Пи методом Монте-Карло

Баллы: 100

Срок сдачи: 17 сентября 2013

## Краткий обзор

На лекции мы познакомились с моделью параллельных вычислений и ее преимуществами, а также как работать с потоками в Java. В этом задании вам предстоит написать параллельную программу, которая вычисляет значение числа Пи. Метод вычисления очень простой:



- Площадь квадрата единичной длины равна 1
- Площадь сектора  $90^\circ$  единичной окружности:  $\pi/4$
- «Бросаем» огромное<sup>1</sup> количество случайных точек в единичный квадрат
- Считаем количество точек, попавших в пределы окружности, т.е. расстояние от которых до (0,0) меньше или равно 1
- Доля точек попавших в окружность равна приближенному значению  $\pi/4$

## Детали реализации

Ваша задача написать параллельную реализацию (ParallelMonteCarloPi.java). При написании программы следуйте инструкциям:

- Первым и единственным входным аргументом программы является количество потоков
- В результате программа выводит следующие данные:

```
PI is 3.14221
THREADS 8
ITERATIONS 1,000,000,000
TIME 12.83ms
```

## Вопросы

Кроме написания программы и отправки результата подумайте над следующими вопросами:

- Как влияет количество итераций (брошенных точек) на конечный результат?
- При одинаковом количестве точек и случайном зерне, как влияет на результат разное количество потоков?
- Как количество потоков влияет на производительность вашей программы? (Для явных результатов вам вероятно понадобится гораздо большее количество сэмплов (итераций))

## Задача 2. Параллельные вычисления математических выражений.

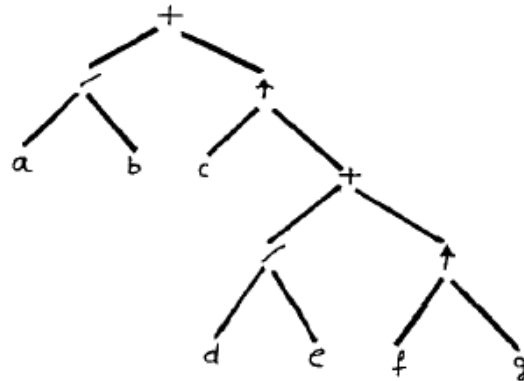
---

<sup>1</sup> Под огромным количеством подразумеваем порядок  $10^7$ - $10^9$

На вход калькулятор принимает математическое выражение. Задача – посчитать значение математического выражения.

В результате разбора математического выражения должно быть получено дерево выражения.

Например: дерево для выражения  $a / b + c \uparrow (d / e + f \uparrow g)$  выглядит следующим образом



Поддерживаемые операции: + - / \* , скобки (), тригонометрия.

Примеры выражений:

$((1+2)*3)/4$

Для реализации параллельных вычислений может быть использован fork / join из Java 7 (<http://habrahabr.ru/blogs/java/128985/>, <http://www.javabeat.net/2012/06/simple-introduction-to-fork-join-framework-in-java-7/>)

### Задача 3. Параллельная запись в БД.

Реализовать запись 10 000 строк в реляционную БД (в одну и ту же таблицу), из 10 потоков одновременно.

Использовать JDBC для записи в БД.

Обратите внимание, что экземпляр класса Connection из JDBC не является thread safe и следует избегать использования одного и того же экземпляра Connection в параллельно работающих threads.