



# Кэширование

Денис Еремеев  
разработчик



# Введение. Вычисление числа Фибоначчи

Классический код

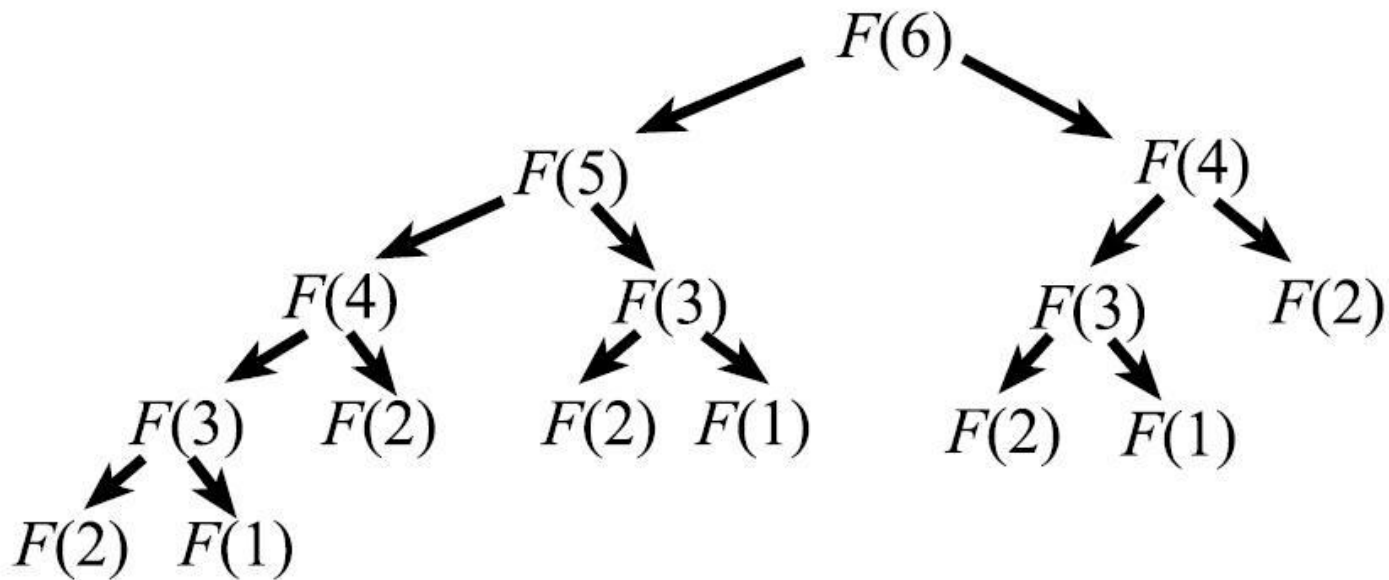
```
public final class Fibonacci {  
  
    public static long calc(long n) {  
        if (n < 2) {  
            return n;  
        } else {  
            return calc(n - 1) + calc(n - 2);  
        }  
    }  
}
```



# Введение. Вычисление числа Фибоначчи

Индекс числа Фибоначчи	Время вычисления, сек
0	0,001
5	0,004
10	0,005
20	0,008
40	0,554
50	49,315
51	78,631

## Введение. Вычисление числа Фибоначчи





# Введение. Вычисление числа Фибоначчи

Оптимизированный код

```
public final class Fibonacci {  
    public static long calc(long n) {  
        long a = 0L;  
        long b = 1L;  
        for (long i = 0; i < n - 1; i++) {  
            long c = a + b;  
            a = b;  
            b = c;  
        }  
        return b;  
    }  
}
```



# Введение. Вычисление числа Фибоначчи

Индекс числа Фибоначчи	Время вычисления, сек
0	0,001
5	0,004
10	0,005
20	0,008
40	0,008
50	0,009
51	0,009

---

# Кэширование. Определение

Кэширование - обмен скорости на память





# Кэширование. Что стоит кэшировать

Что кэшировать?

- Результаты вычислений часто повторяющихся запросов;
- Соединения, потоки. Например, `connection pool` для бд, почтового сервера.





# Кэширование. Что стоит кэшировать. Что не надо

Что кэшировать?

- Результаты вычислений часто повторяющихся запросов;
- Соединения, потоки. Например, connection pool для бд, почтового сервера.

Что кэшировать не нужно?

- Данные, запрос которых маловероятен;
- Часто редактируемые (обновляемые) данные.



# Кэширование. Виды

Локальный кэш.

Плюсы:

- Простота
- Скорость
- Надежность.



# Кэширование. Виды

Локальный кэш.

Плюсы:

- Простота
- Скорость
- Надежность.

Минусы:

- Невозможность масштабирования
- Нагрузка на память. Вызов сборщика мусора.



# Кэширование. Виды

Внешний кэш.

Плюсы:

- Масштабирование.



# Кэширование. Виды

Внешний кэш.

Плюсы:

- Масштабирование.

Минусы:

- Накладные расходы на сетевое взаимодействие и сериализацию.

\* Расходы на сериализацию решаются выбором подходящего сериализатора.

---

## Кэширование. Инвалидация





# Кэширование. Инвалидация

Инвалидация кэша:

- Сбрасывать по ключу



# Кэширование. Инвалидация

Инвалидация кэша:

- Сбрасывать по ключу
- Сбрасывать весь кэш





# Кэширование. Инвалидация

Инвалидация кэша:

- Сбрасывать по ключу
- Сбрасывать весь кэш
- Кэшировать на время



# Кэширование. Инвалидация

Инвалидация кэша:

- Сбрасывать по ключу
- Сбрасывать весь кэш
- Кэшировать на время
- Неинвалидировать )



# Кэширование. Оценка эффективности

Эффективность кэширования:

- Производительность



# Кэширование. Оценка эффективности

Эффективность кэширования:

- Производительность
- Количество попаданий\промахов

---

## Кэширование. Типичные ошибки





# Кэширование. Типичные ошибки

Типичные ошибки:

- Кэш как база данных



# Кэширование. Типичные ошибки

Типичные ошибки:

- Кэш как база данных
- Кэш каккрытие неоптимальности работы системы



# Кэширование. Выводы

Нельзя просто так взять и применить кэш. Нужно:

- Помнить, что кэш - не БД
- Помнить, что кэш не спрячет костыли
- Выбрать что кэшировать
- Выбрать вид кэша - локальный или внешний
- Выбрать способ\способы инвалидации кэша
- Всегда и еще раз всегда мониторить работу с кэшем. Производить оценку



---

# Спасибо за внимание!

Требуйте носки Demix



в кассах магазинов Спортмастер