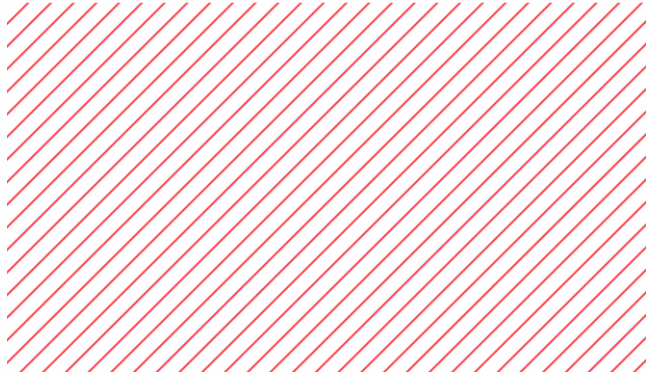
академия больших данных

mail.ru group

Хеш-таблицы

Шовкопляс Григорий

Введение в алгоритмы и структуры данных



Хеш-таблицы

Дисклеймер

- В данной лекции будет несколько казуальных вилами писанных доказательств, потому что продвинутая теория вероятностей это сложно…

Зачем нужны хеш-таблицы?

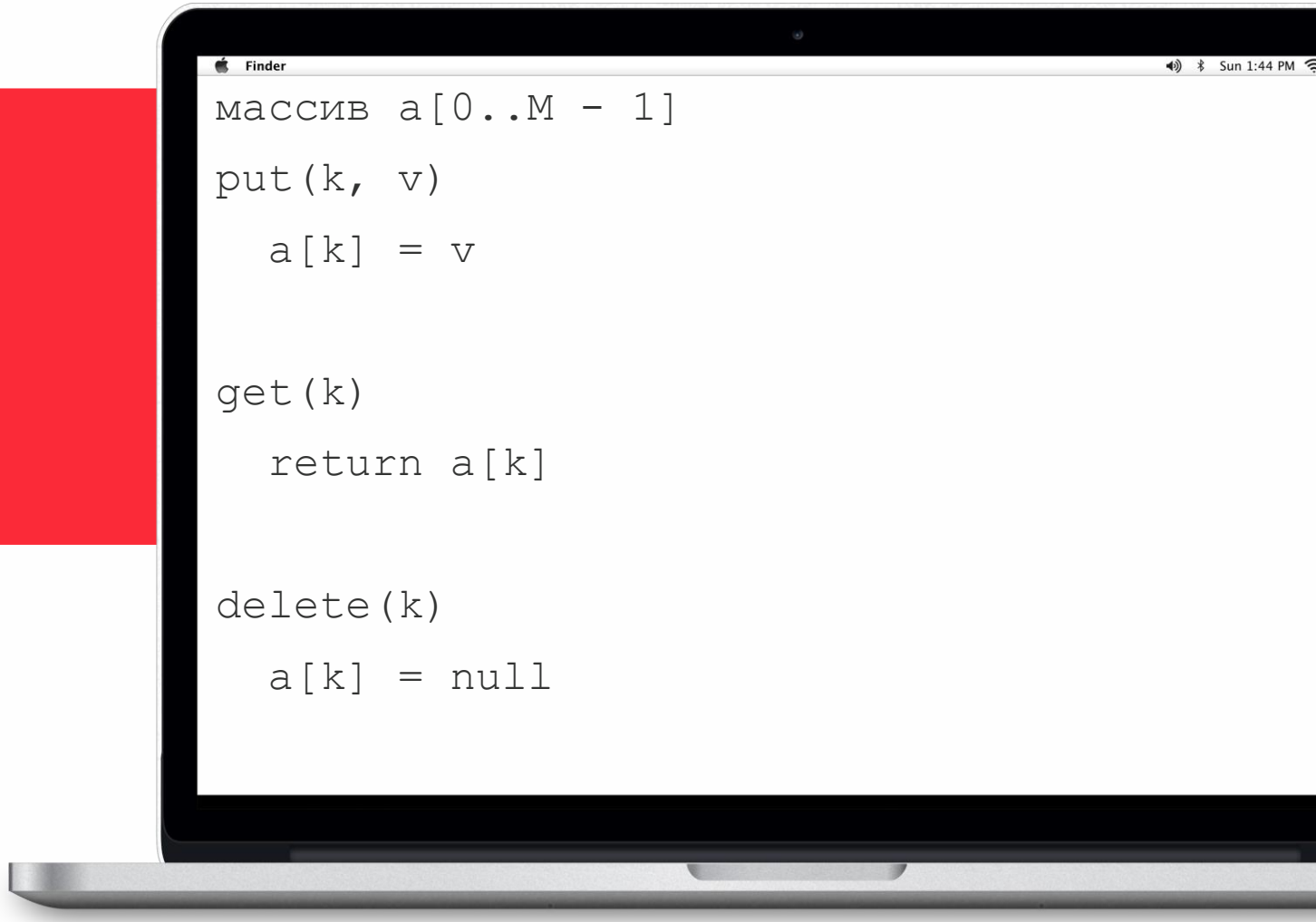
Применение хеш-таблиц

- Для абстрактных структур данных set и map
- Set (множество):
  - insert(x)
  - delete(x)
  - contains(x)
- Map (ассоциативный массив):
  - put(k, v)
  - delete(k)
  - get(k)

Некоторая реализация Map

$0 \le k \le M - 1$

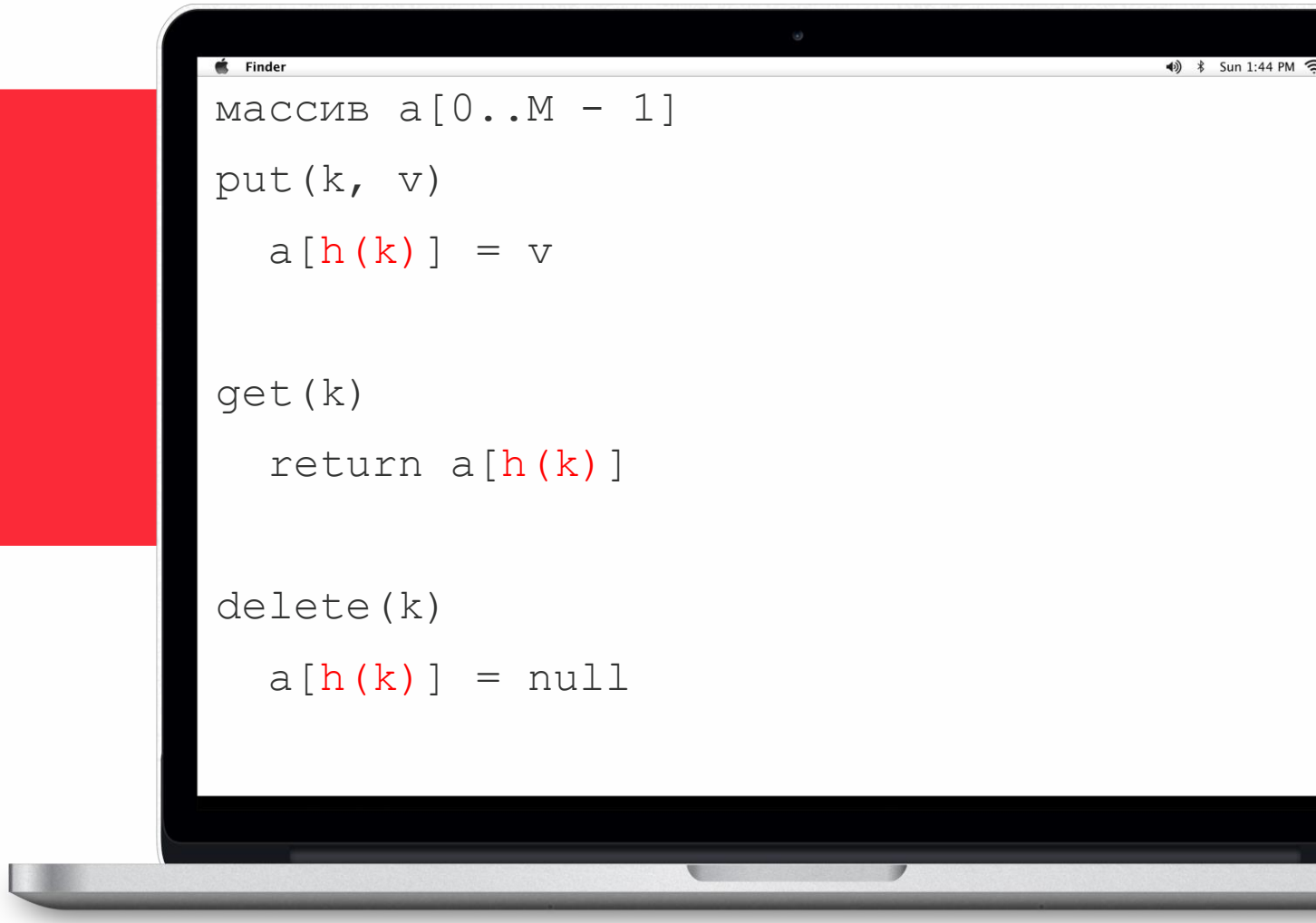
M – маленькое



Некоторая реализация Map

$0 \le k \le U - 1$
U – большое

M – маленькое
функция $h(k): [0..U) \to [0..M)$



Некоторая реализация Map

- Что может пойти не так?
- Коллизия: $x \neq y : h(x) = h(y)$
- Почему коллизии вообще бывают?
- Хорошая хеш-функция: случайная хеш-функция
  - Все немного сложнее, но сегодня считаем так
- Если хеш-функция хорошая, то вероятность коллизии маленькая?

Вероятность коллизии

- Есть массив размера n
- Какое минимальное число элементов нужно запихать туда, чтобы получить коллизию?
- С вероятностью > 0?
  - два
- С вероятностью < 50%?
  - $\sqrt{n}$
  - Парадокс дней рождений

Парадокс дней рождения

- Если есть группа из хотя бы 23 человек
- Вероятность совпадения дней рождения у двух из них > 50%
- Если группа на 60 человек, то больше 99%
- Казуально: 253 пары людей, вероятность того, что совпадет внутри хотя бы одной пары большая
- $p(n) - \text{вероятность, что у всех будут различные даты ДР}$
  - $p(n) = 1 \times \left(1 - \frac{1}{365}\right) \times \left(1 - \frac{2}{365}\right) \ldots \times \left(1 - \frac{n-1}{365}\right) =$
  - $= \frac{365 \times 364 \ldots \times (365-n+1)}{365^n} = \frac{365!}{365^n (365-n)!}$

Парадокс дней рождения

Вероятность коллизии:

$$1 - p(n)$$

n	1 - p(n)
10	12%
20	41%
30	70%
50	97%
100	99. 99996%
200	99.99999999999999999999999998%
300	$(1 - 7 \times 10^{-73}) \times 100\%$
350	$(1 - 3 \times 10^{-131}) \times 100\ \%$
366	100%

Вероятность коллизии

- Вывод: коллизии это нормально
- Надо научится их обрабатывать

Метод цепочек

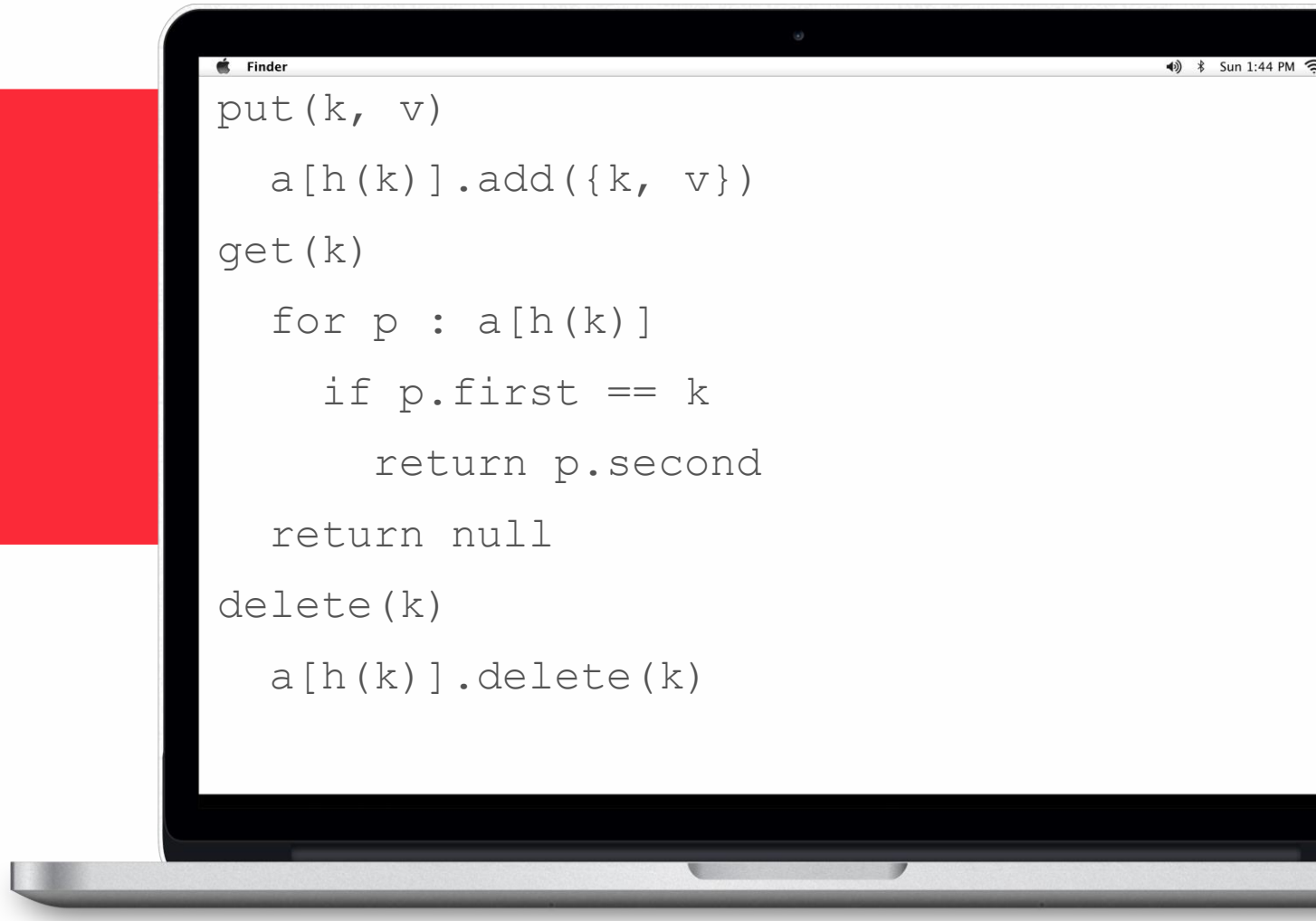
Метод цепочек

- Каждая ячейка массива, теперь список
- В случае коллизии храним по одному хешу несколько элементов

Метод цепочек

Тупенькая реализация Map

Какие проблемы?



Метод цепочек

- Оценим время работы (казуально)
- Худший случай O(n)
- В среднем O(n/M)
  - $n \sim M \rightarrow O(1)$

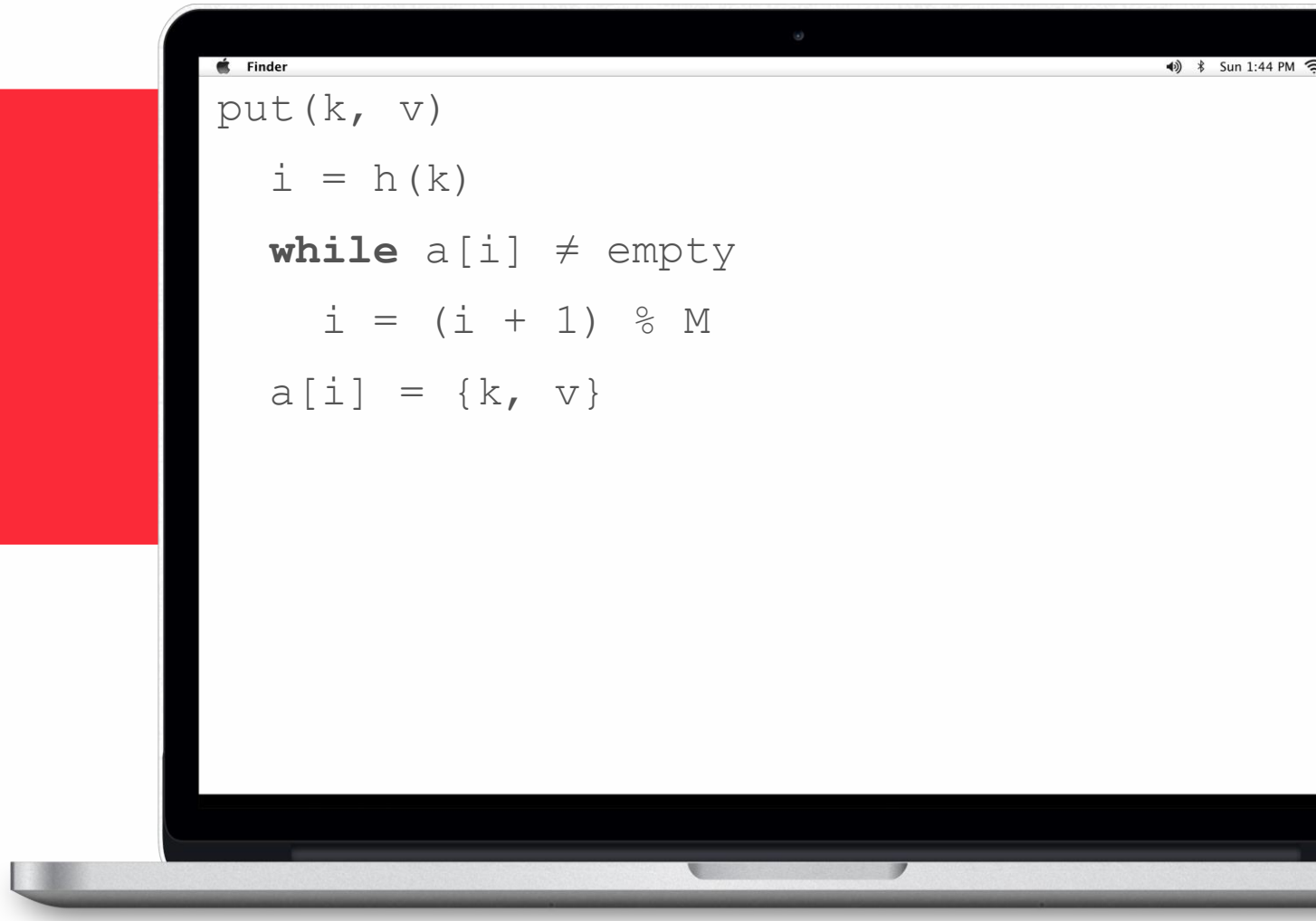
Открытое хеширование (адресация)

Открытое хеширование

- Списки не очень эффективны и хранить сложно
- Давайте обойдемся обычным массивом
- Если случилась коллизия идем дальше, пока не встретим пустую ячейку
- Вставим туда

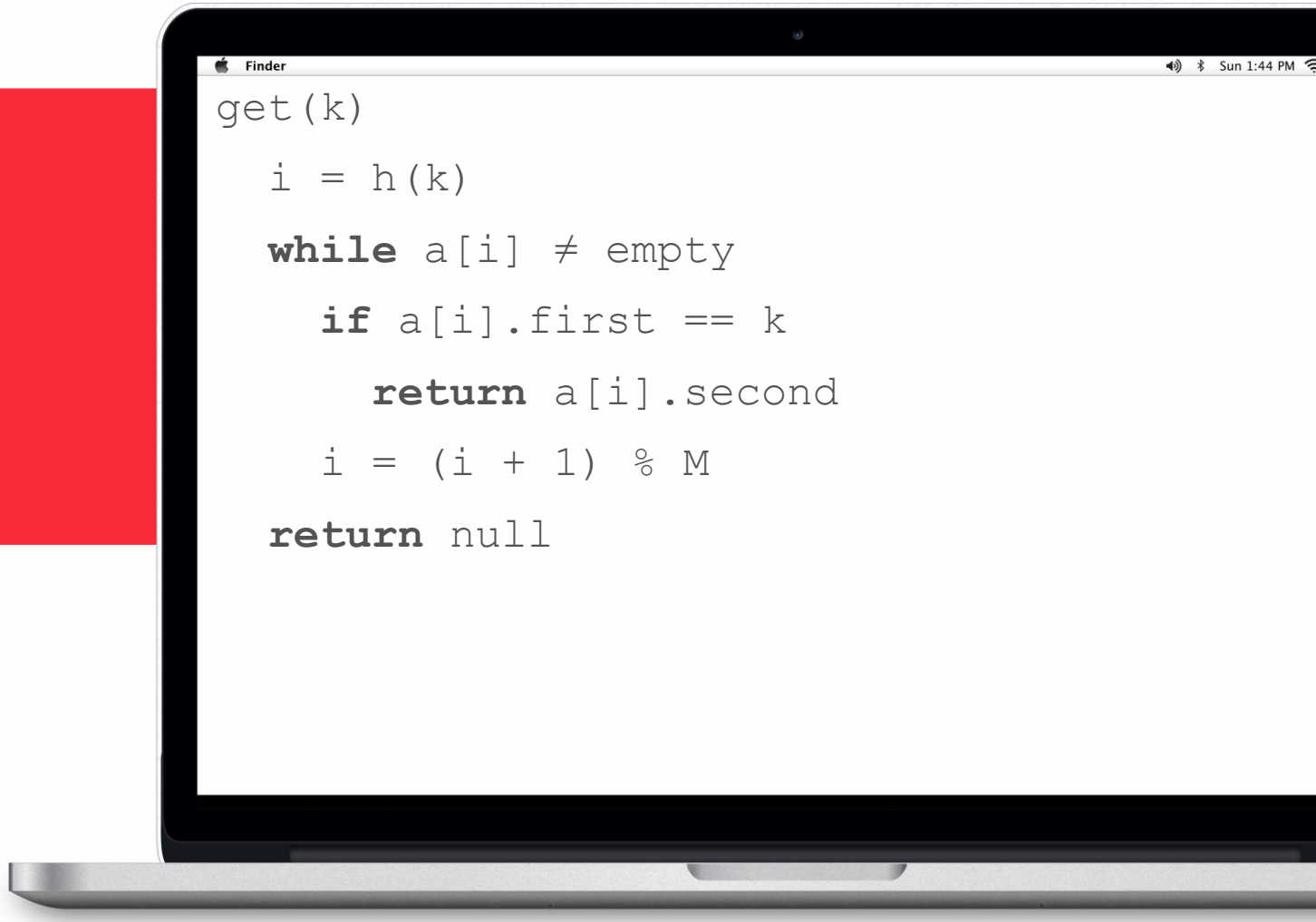
Открытое Хеширование

Реализация Map



Открытое Хеширование

Реализация Map



Открытое хеширование

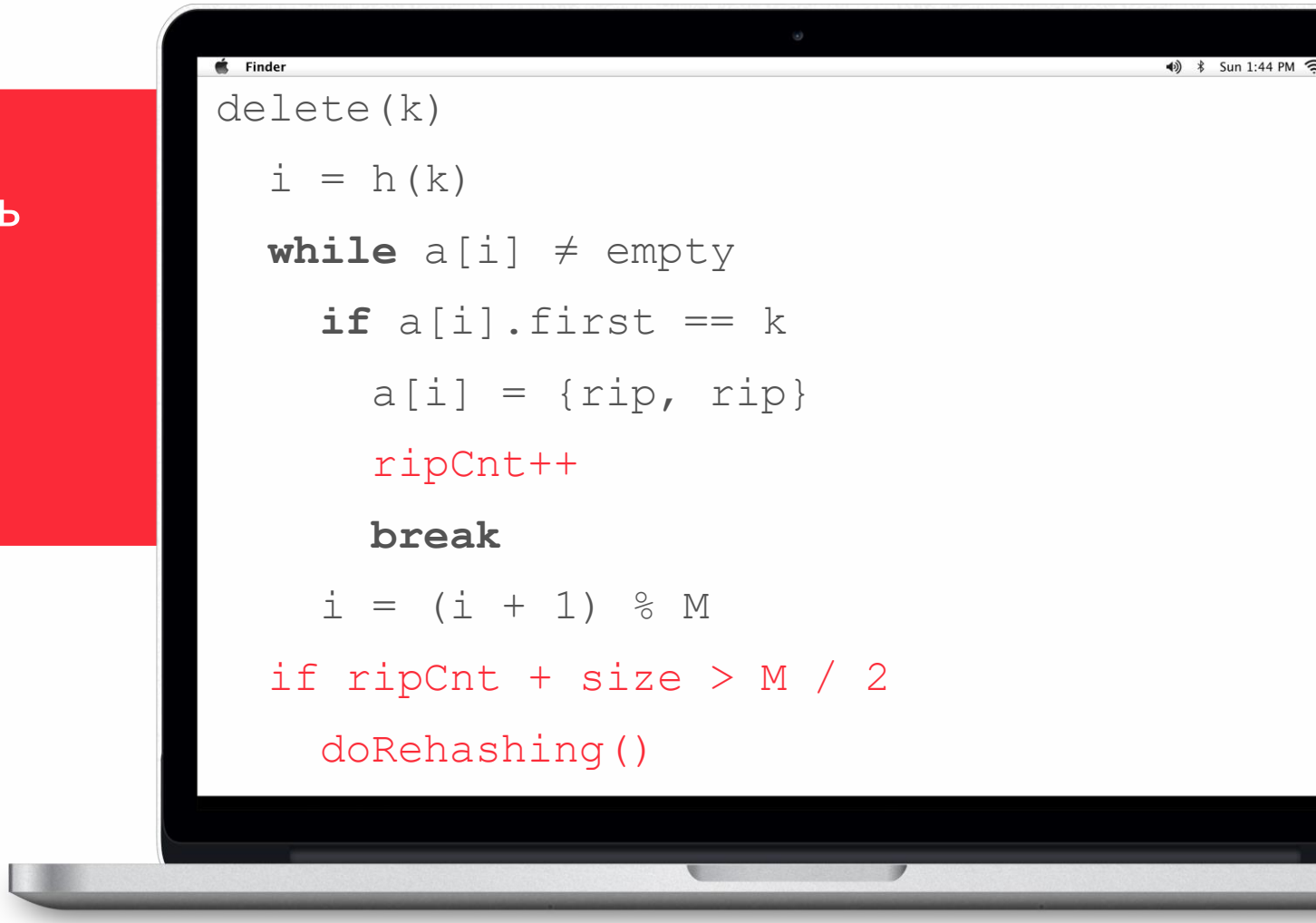
- Удалять научимся позже
- Какого размера выбрать массив, чтобы работало быстро?
- Если M = n, плохо
- Если M = 2n, вероятность, что следующая пустая ½
- Если закончилось место?
  - Расширим массив
  - Поменяем Хеш-функцию и перехешируем
  - Амортизированно O(1), как в векторе

Открытое хеширование

- Почему удалять нетривиально?
- Как тогда удалять?

Открытое Хеширование

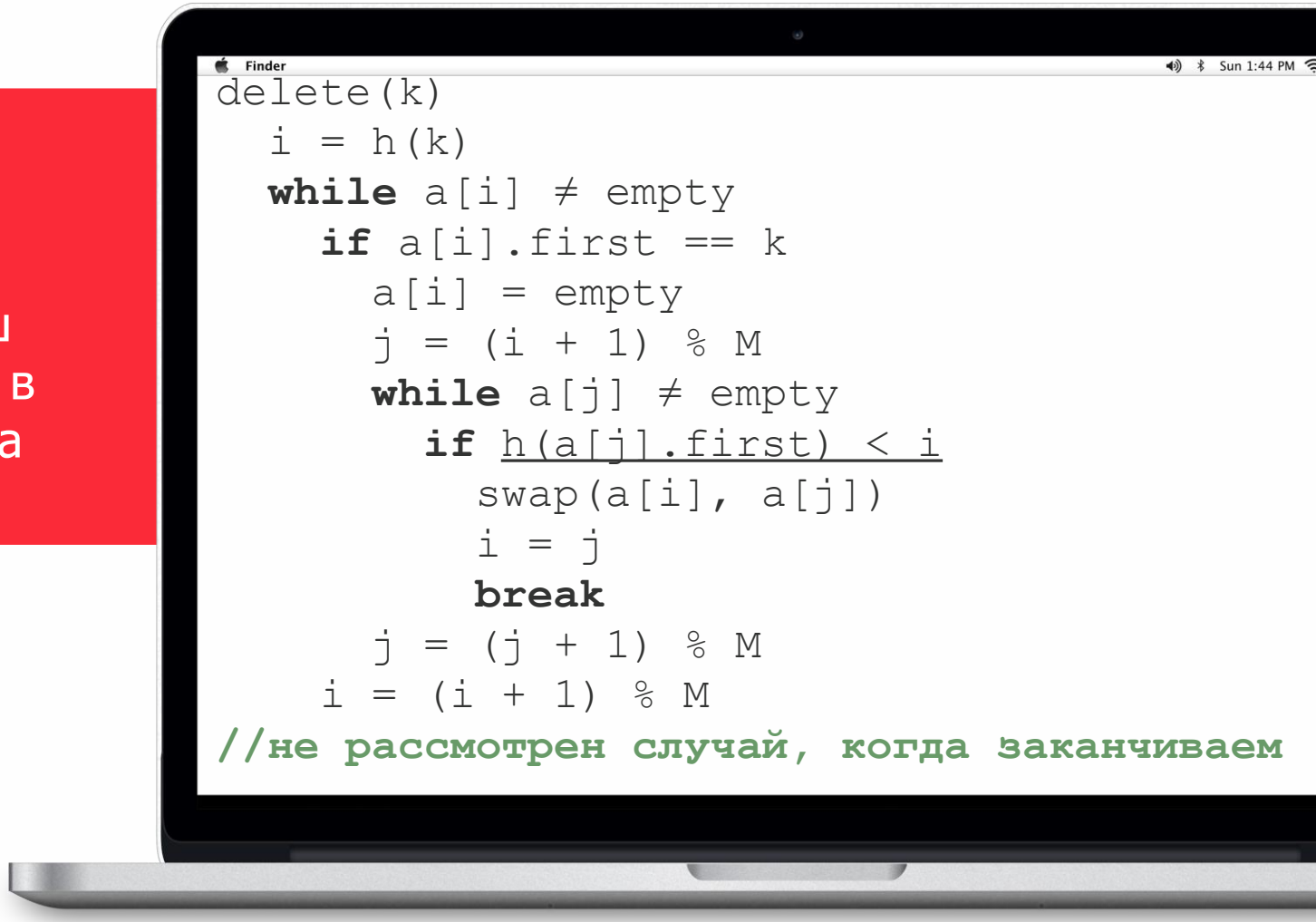
План-капкан, не будем удалять явно



Открытое Хеширование

Как сделать без «заглушек»?

Если правее есть элемент, хеш которого левее перенесем его в дырку, получим новую, и снова




```
delete(k)
  i = h(k)
  while a[i] ≠ empty
    if a[i].first == k
      a[i] = empty
      j = (i + 1) % M
      while a[j] ≠ empty
        if h(a[j].first) < i
          swap(a[i], a[j])
          i = j
          break
      j = (j + 1) % M
    i = (i + 1) % M
//не рассмотрен случай, когда заканчиваем
```

Откуда берутся хорошие хеш-функции

Хеш-функции для чисел

- Случайных хеш-функций не бывает
- Бывают псевдо-случайные
- Семейство хеш-функций
- h(x) = (Ax % P) % M
  - P > M и простое
  - A – случайное

Хеш-функции для строк

- Полиномиальный хеш
- $h(S) = ((A^0 s_0 + A^1 s_1 \dots) \% P) \% M$
  - P > M и простое
  - A – случайное (обычно берут простое больше алфавита)
- $h(S) = ((A^{n-1} s_0 + A^{n-2} s_1 \dots) \% P) \% M$
- Почему вероятность коллизии небольшая?

Десерт

Десерт

- Фильтр Блума
- Хеширование кукушки

Все!