

# Bees

F1124004  
5110309084  
蒋舜宁

# What have I done?

- 6/6:
- array
- deque
- splay tree
- separate chaining

# Iterator

---

- Iterator
- ConstIterator

# Utility

- addAll
  - addAll(\*this,a)
- Entry<K,V>
  - const key K
- ElementNotExist()
  - { throw ElementNotExist();}

# ArrayList

---

- seek(index)
- store index or link?

# LinkedList

---

- head tail seek(index)
- use head instead of NULL

# TreeSet

---

- splay – slower, and a waste!!!
- ConstIterator's problem

# TreeMap

- a secondary product of TreeSet
- Entry<K,V>'s problem



# HashSet

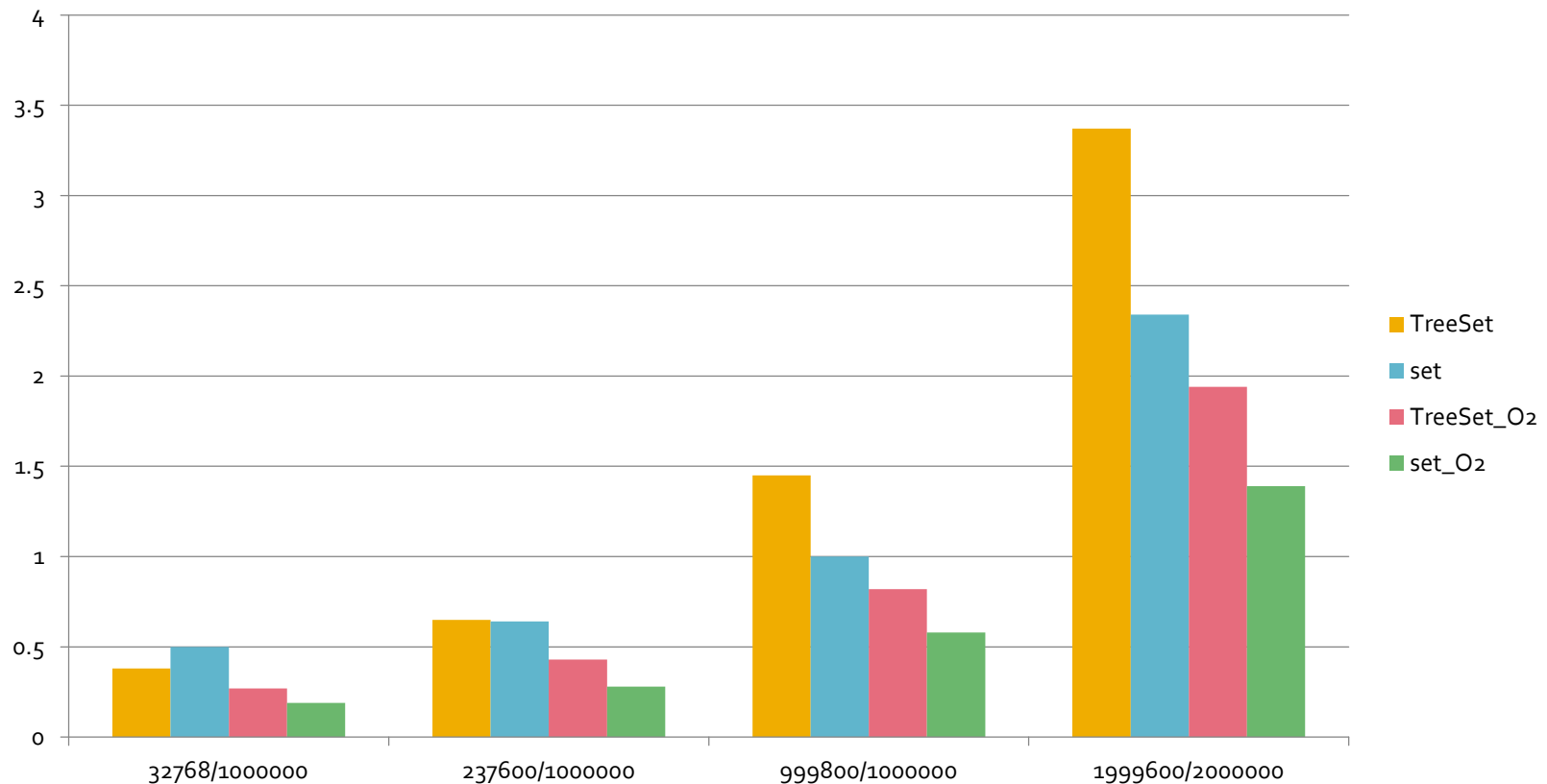
- separate chaining
- a big prime
- an array of LinkedList?

# HashMap

- a secondary product of TreeSet
- No LinkedList with both head and tail.

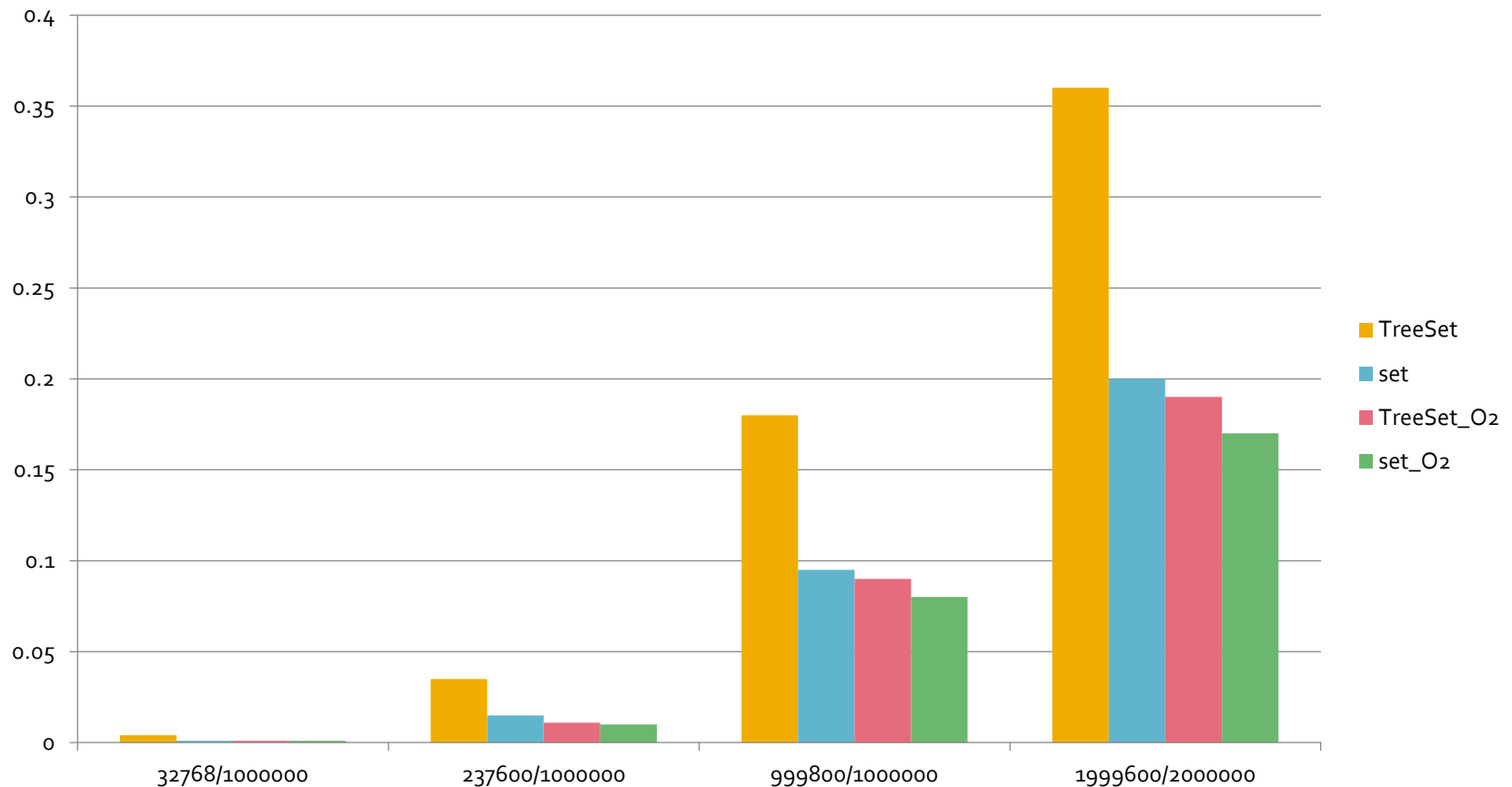
# Comparison - TreeSet & set

- Add  $b$  elements.  $a$  of them are distinct. ( $a/b$ )



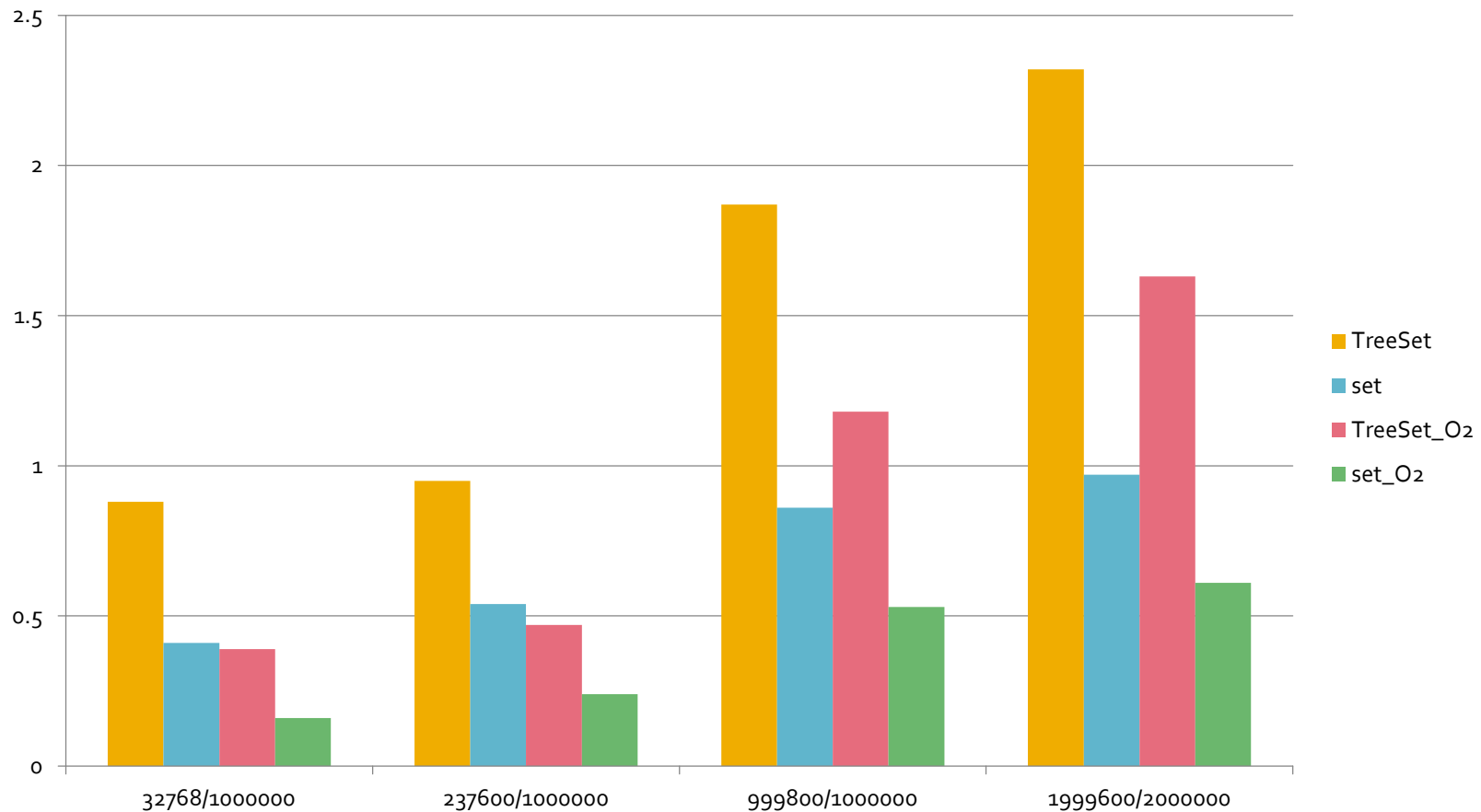
# Comparison - TreeSet & set

- Use ConstIterator to traverse a elements.



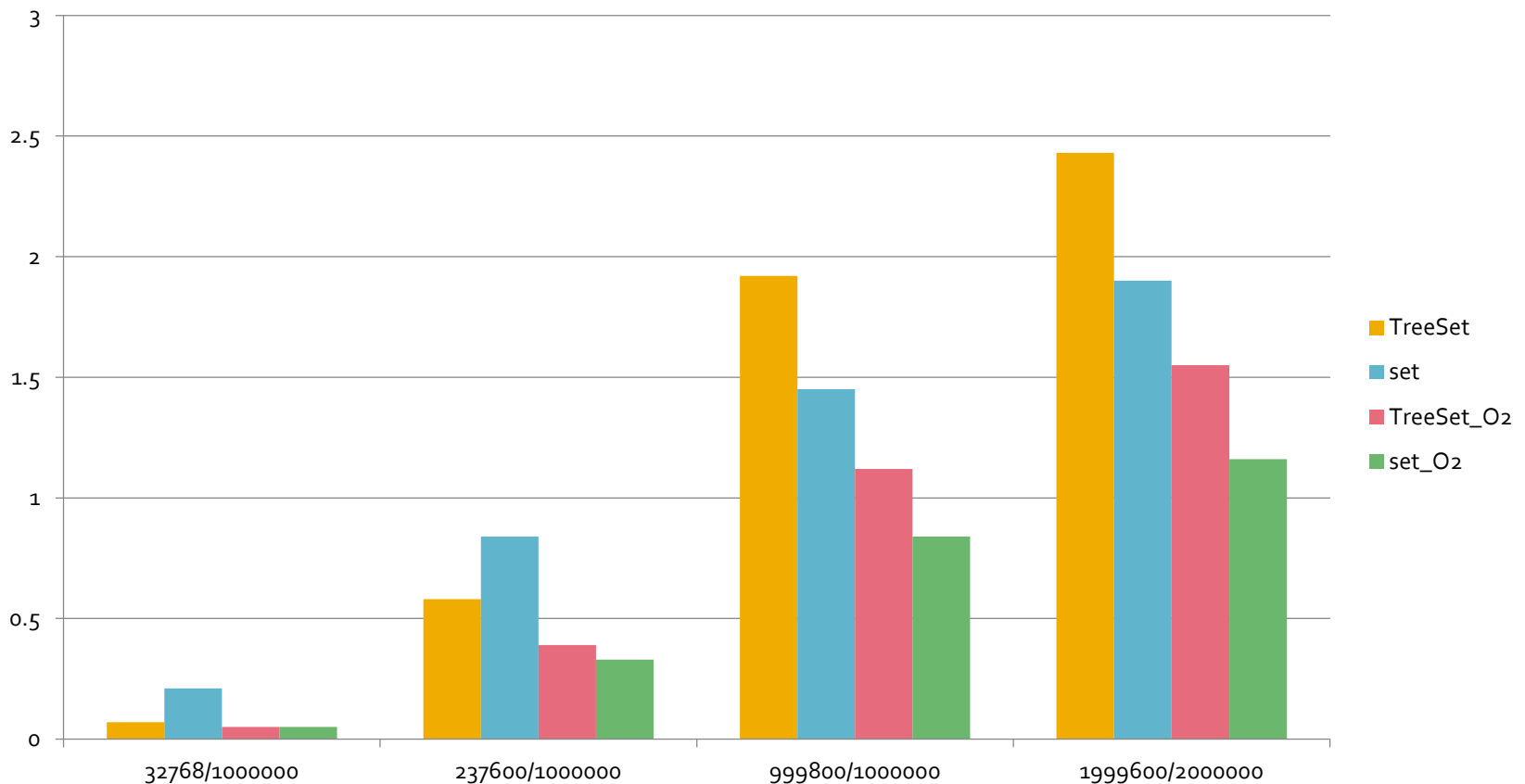
# Comparison - TreeSet & set

- Count 1,000,000 times (guarantee).



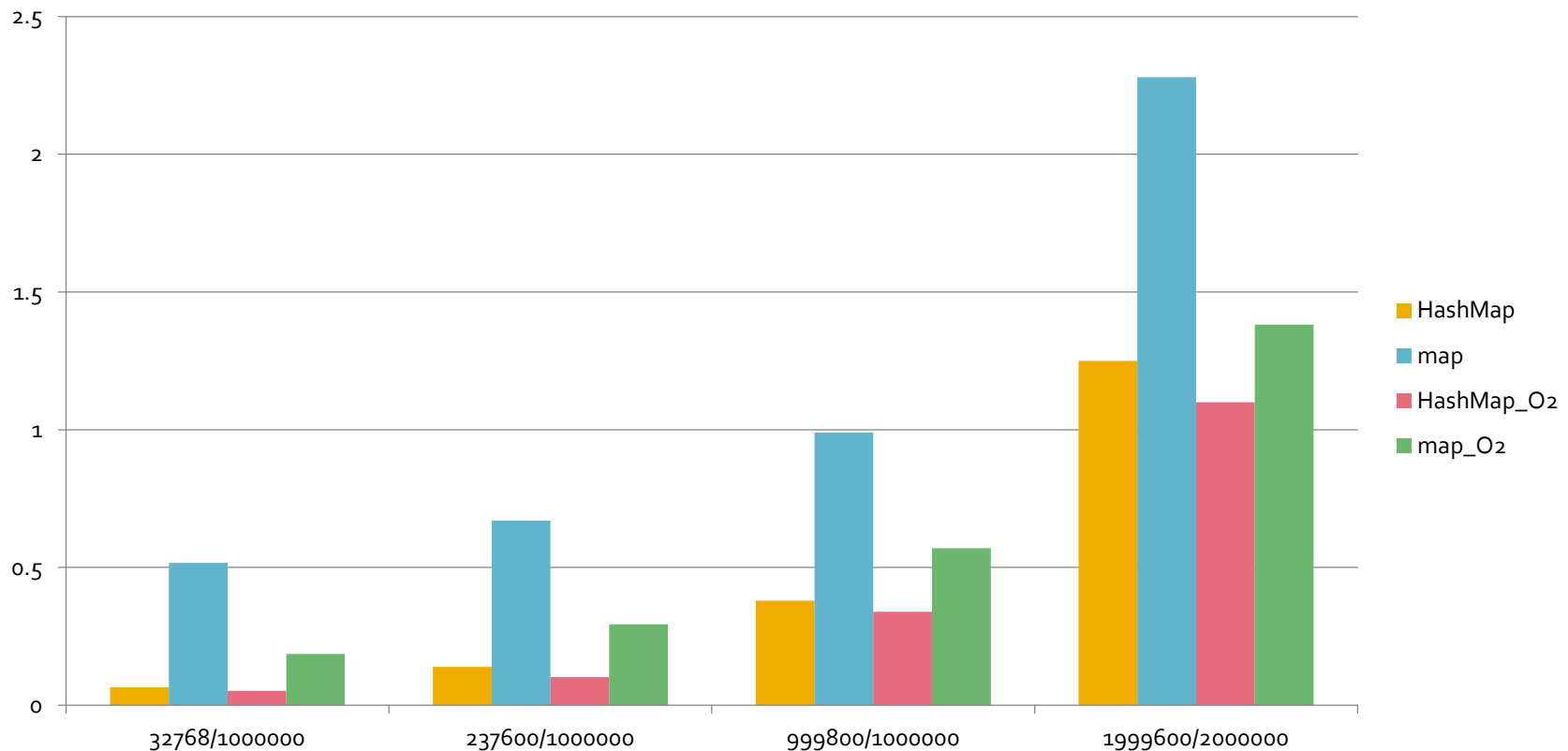
# Comparison - TreeSet & set

- Remove 1,000,000 elements in random order.



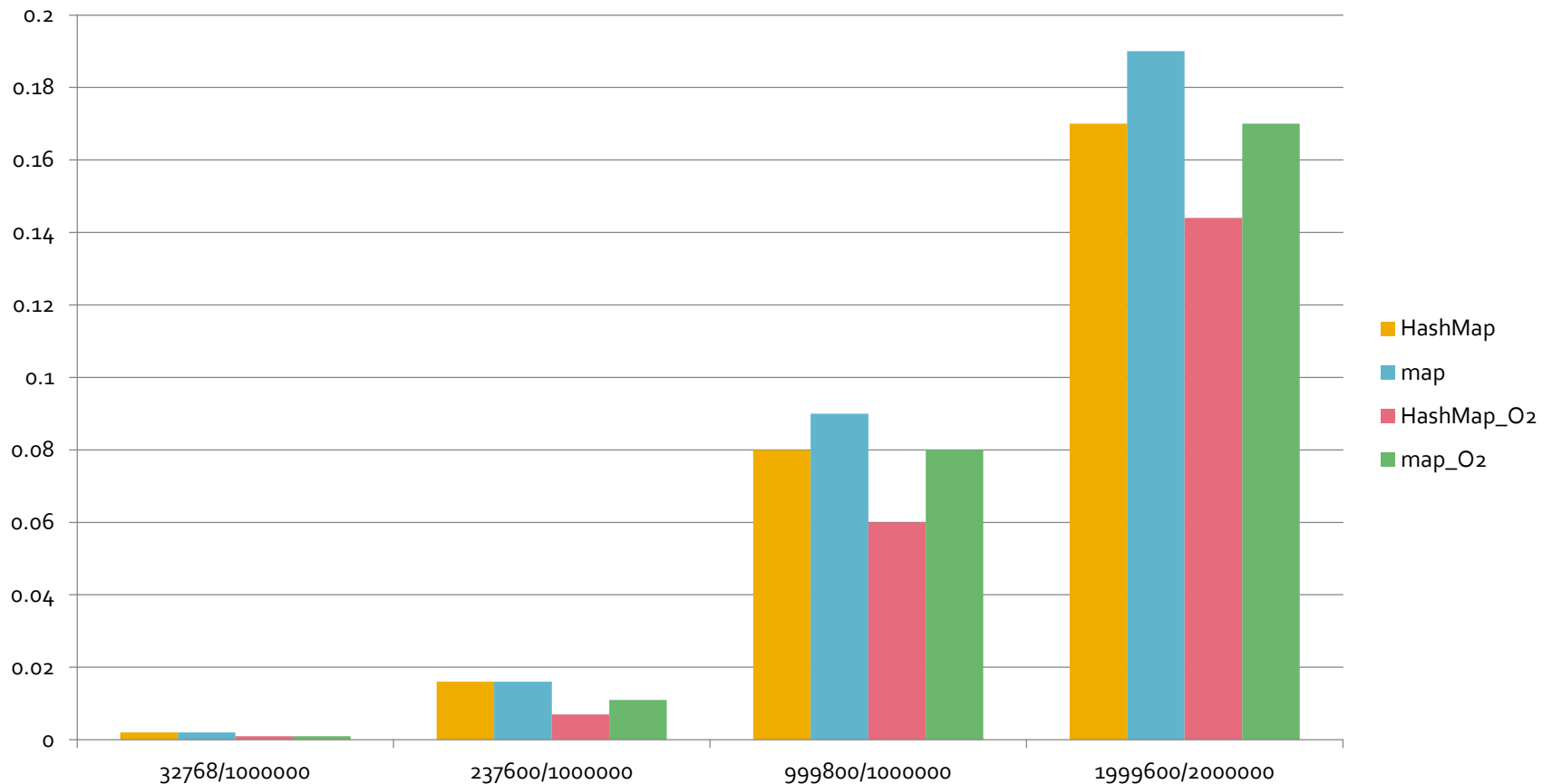
# Comparison – Hashmap & map

- Add  $b$  elements.  $a$  of them are distinct. ( $a/b$ )



# Comparison – Hashmap & map

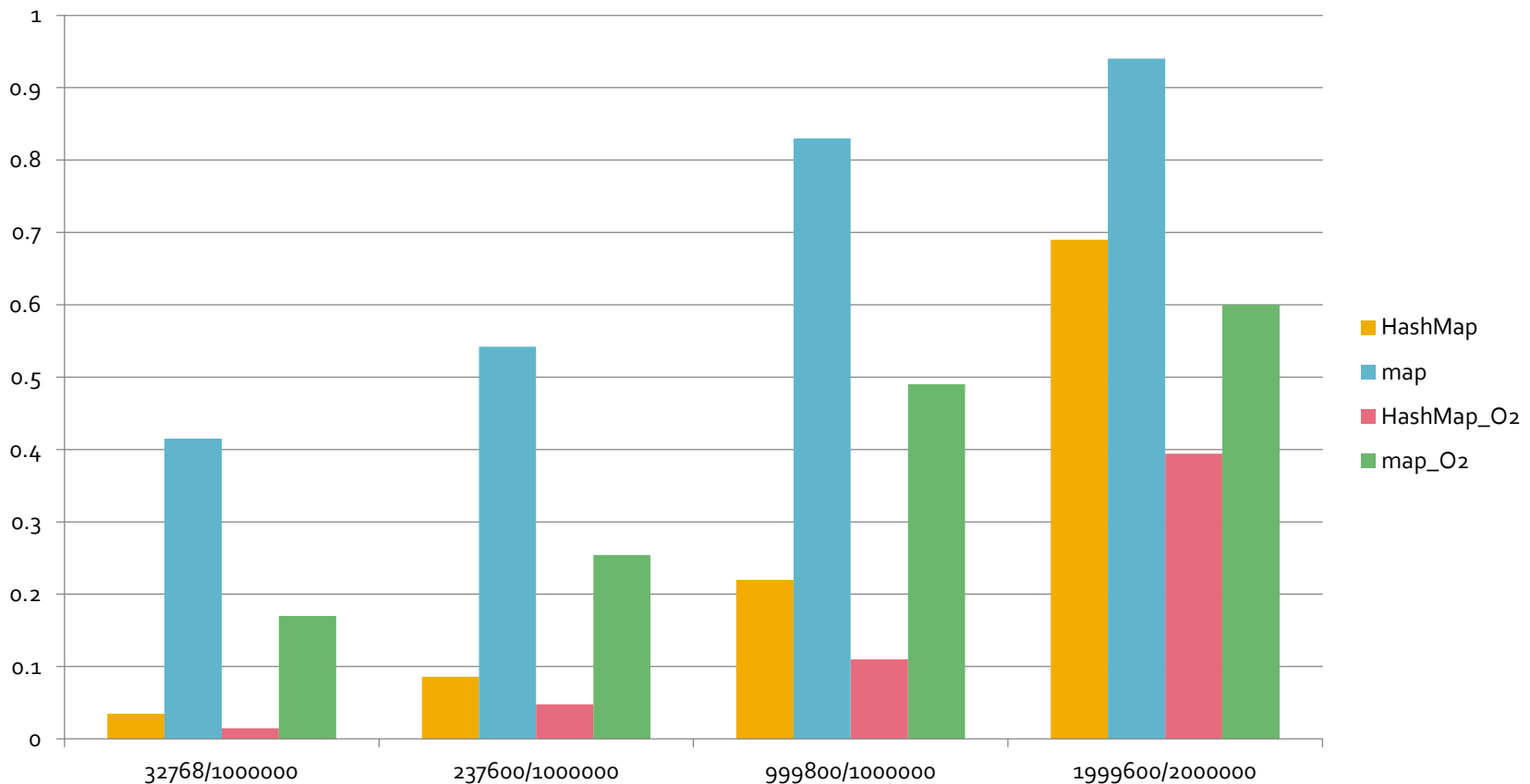
- Use ConstIterator to traverse a elements.





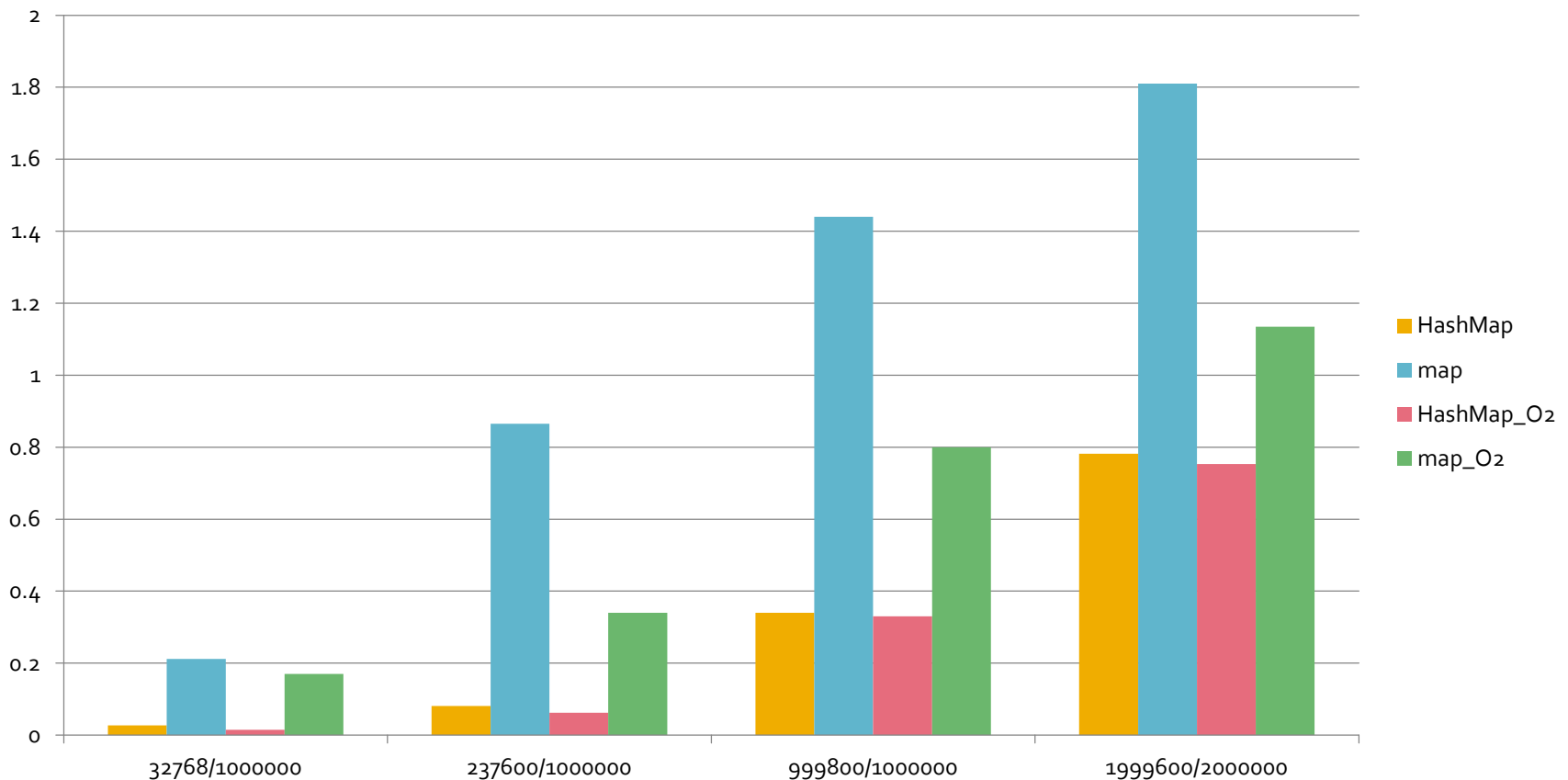
# Comparison – Hashmap & map

- Count by Key 1,000,000 times (guarantee).



# Comparison – Hashmap & map

- Remove 1,000,000 elements in random order.



# Other containers

- ArrayList & vector?
- LinkedList & list?
- HashSet & set?
- TreeMap & map?

# Opinions

- Eliminate?
- Platform?
- Pursue test&ability or pursue speed?

# Thanks!

---

- Q&A