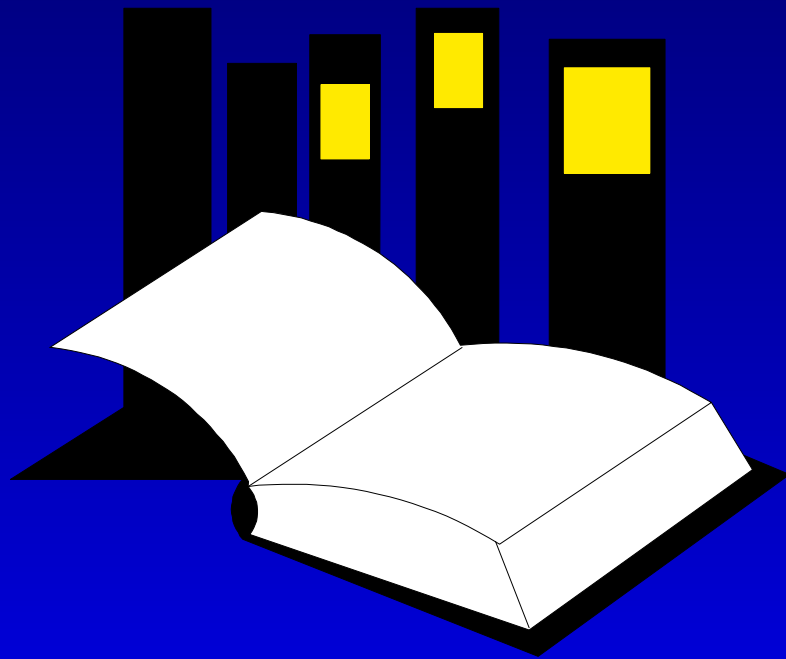
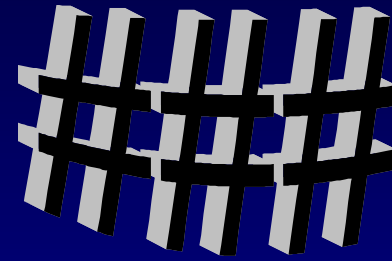


Hash Tables

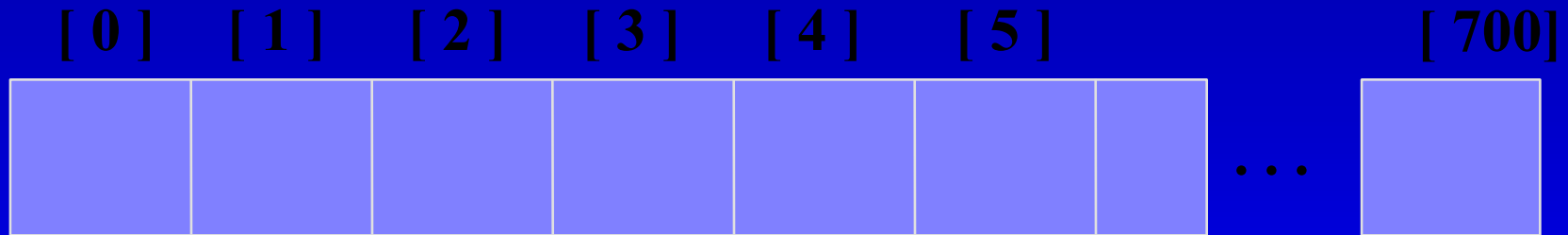


- ❑ We discussed several ways of storing information in an array, and later searching for the information.
- ❑ Hash tables are a common approach to the storing/searching problem.
- ❑ This presentation introduces hash tables.

**Data Structures
and Other Objects**

What is a Hash Table ?

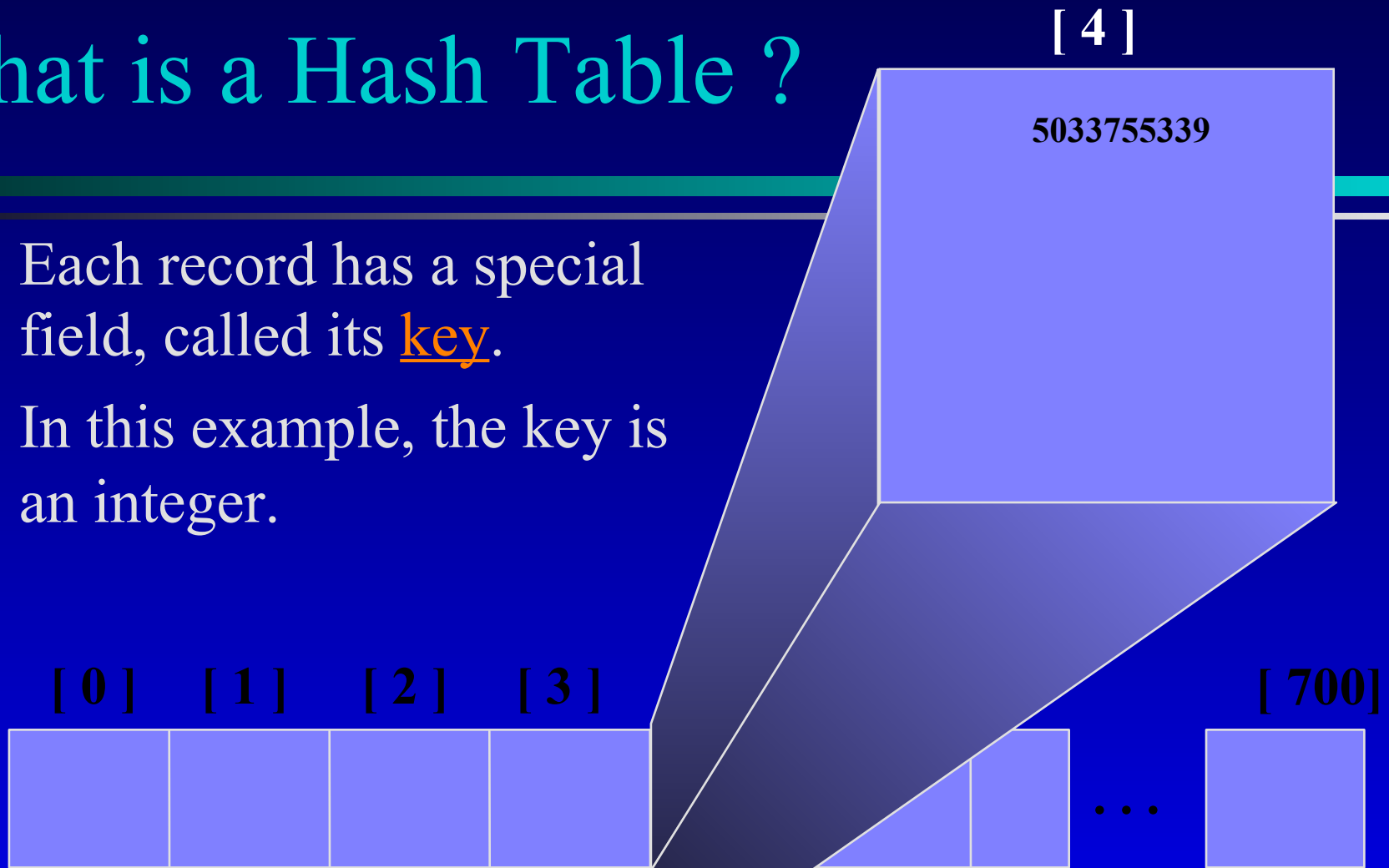
- ❑ The simplest kind of hash table is an array of records.
- ❑ This example has 701 records.



An array of records

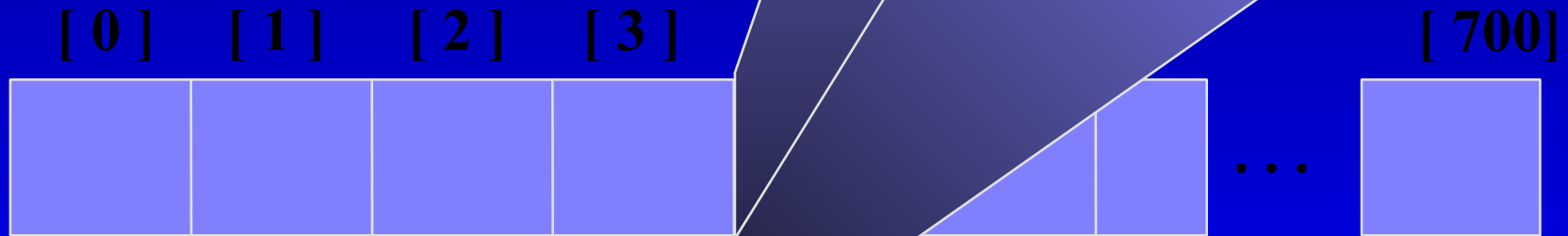
What is a Hash Table ?

- Each record has a special field, called its key.
- In this example, the key is an integer.



What is a Hash Table ?

- The number might be a person's campus phone number, and the rest of the record has information about the person.



What is a Hash Table ?

- When a hash table is in use, some spots contain valid records, and other spots are "empty".



Inserting a New Record

- ❑ In order to insert a new record, the **key** must somehow be **converted to** an array **index**.
- ❑ The index is called the **hash value** of the key.

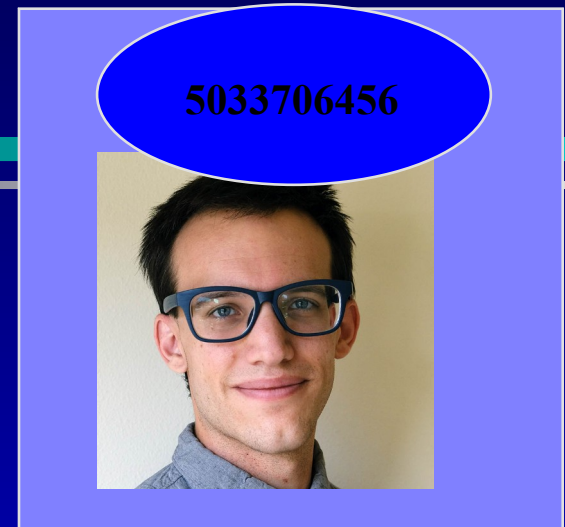


Inserting a New Record

- Typical way create a hash value:

(Number mod 701)

What is $(580625685 \bmod 701)$?



Inserting a New Record

- Hash with modular arithmetic:

(Phone# mod 701)

What is $(5033706454 \bmod 701)$?



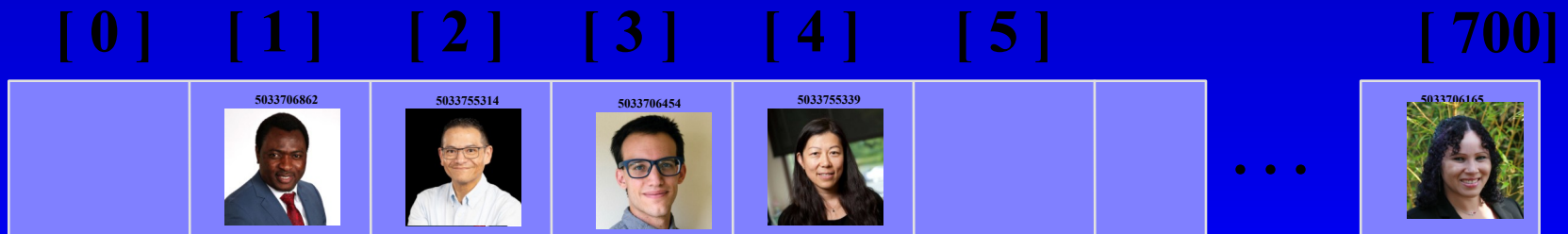
Inserting a New Record

- The hash value is used for the location of the new record.



Inserting a New Record


- The hash value is used for the location of the new record.



Collisions

- Here is another new record to insert, with a hash value of 2.

5033706453



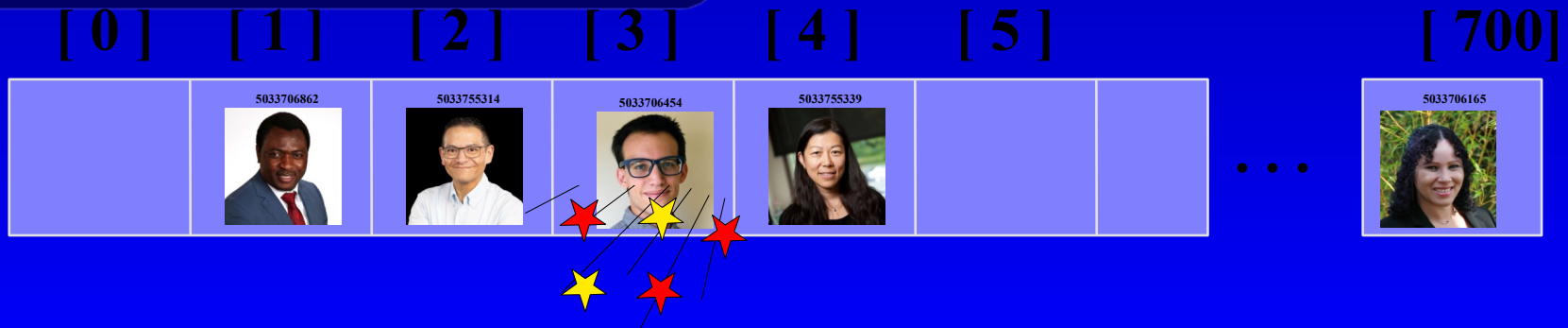
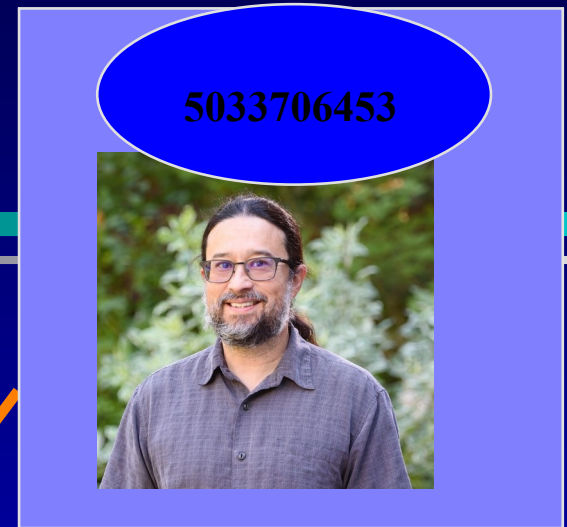
My hash value is [2].



Collisions

- This is called a **collision**, because there is already another valid record at [2].

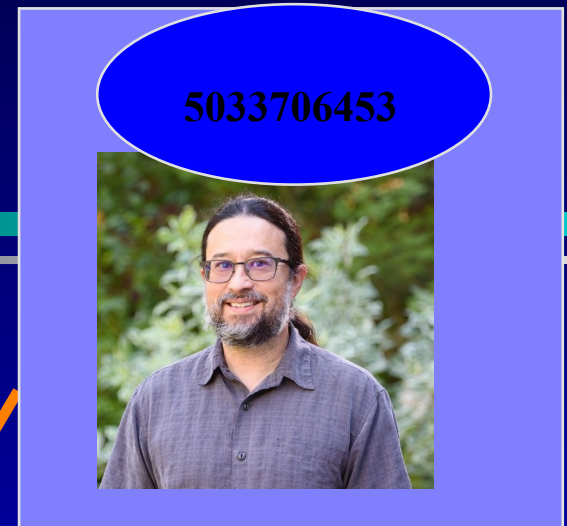
When a collision occurs, move forward until you find an empty spot.



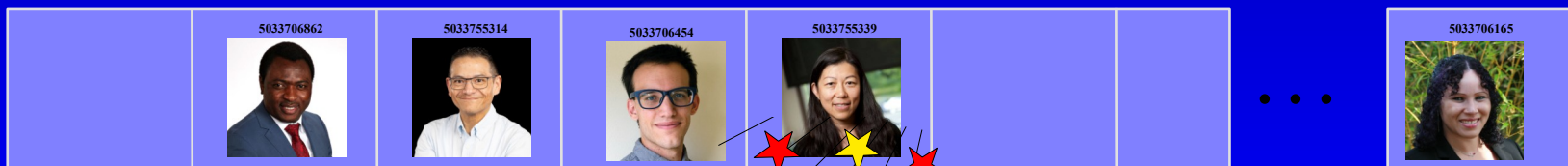
Collisions

- This is called a **collision**, because there is already another valid record at [2].

When a collision occurs, move forward until you find an empty spot.



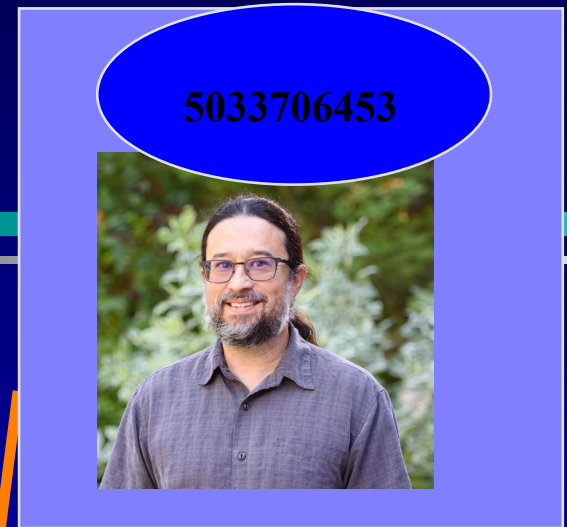
[0] [1] [2] [3] [4] [5] ... [700]



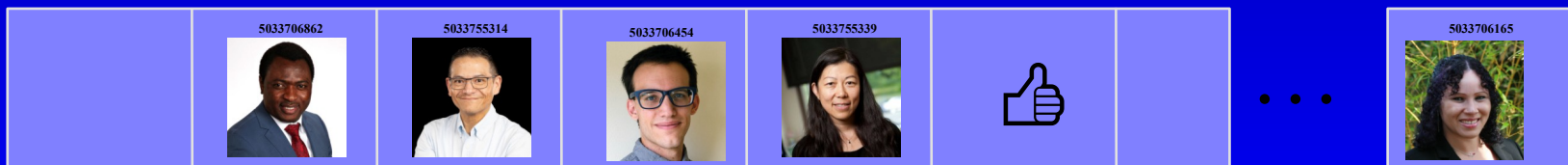
Collisions

- This is called a **collision**, because there is already another valid record at [2].

When a collision occurs,
move forward until you
find an empty spot.



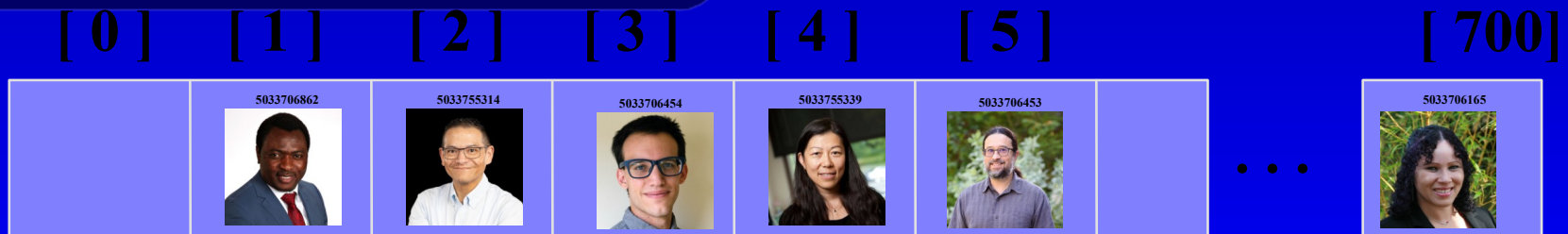
[0] [1] [2] [3] [4] [5] ... [700]



Collisions

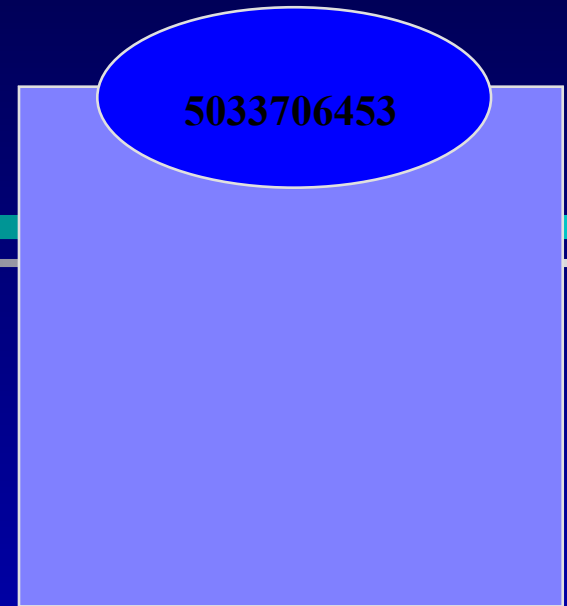
- This is called a **collision**, because there is already another valid record at [2].

The new record goes
in the empty spot.

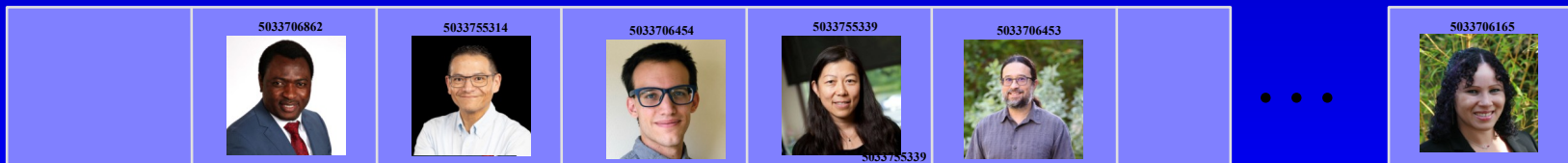


Searching for a Key

- The data that's attached to a key can be found fairly quickly.

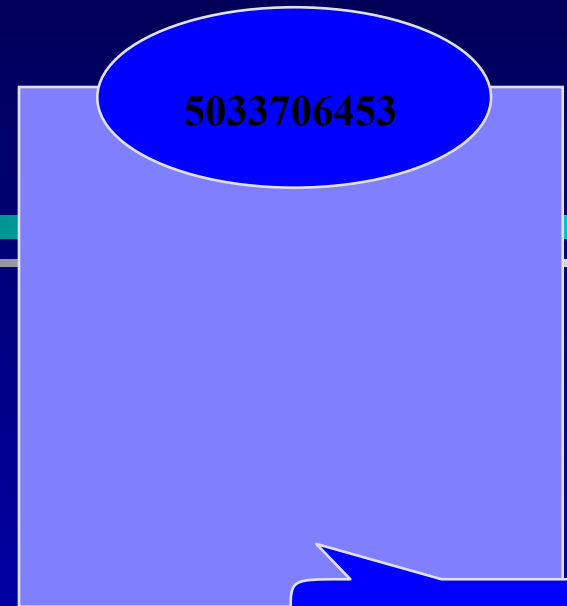


[0] [1] [2] [3] [4] [5] ... [700]



Searching for a Key

- ❑ Calculate the hash value.
- ❑ Check that location of the array for the key.



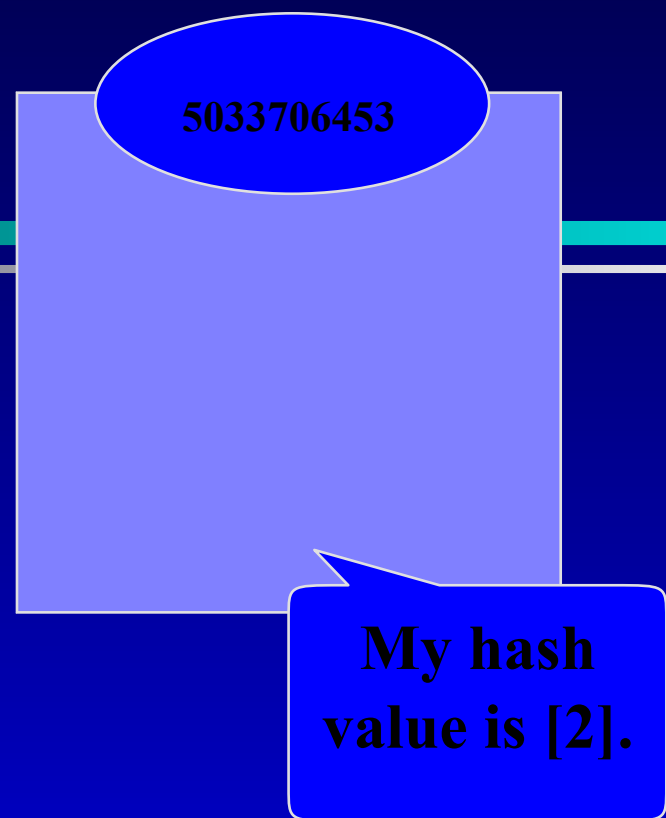
My hash value is [2].

Not me.



Searching for a Key

- Keep moving forward until you find the key, or you reach an empty spot.

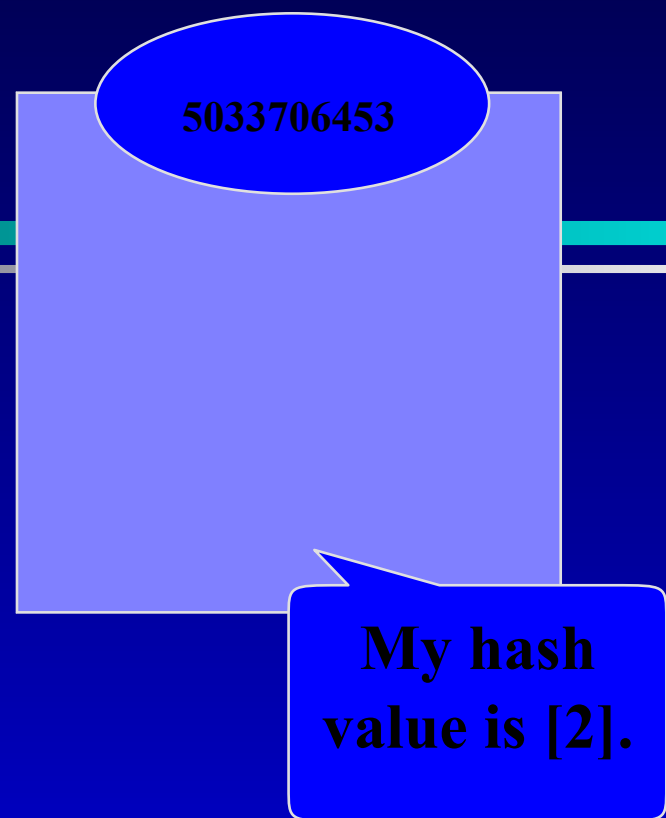


Not me.



Searching for a Key

- Keep moving forward until you find the key, or you reach an empty spot.



Not me.



Searching for a Key

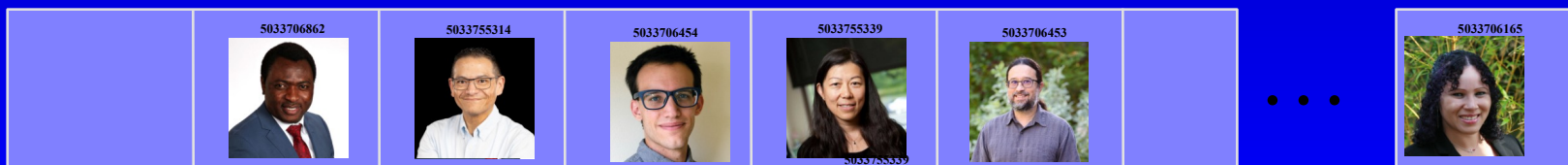
- Keep moving forward until you find the key, or you reach an empty spot.



My hash value is [2].

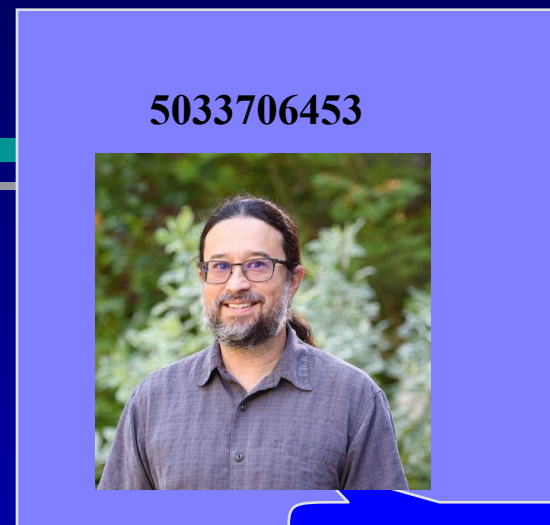
Yes!

[0] [1] [2] [3] [4] [5] ... [700]



Searching for a Key

- When the item is found, the information can be copied to the necessary location.



My hash value is [2].

Yes!



Deleting a Record

- Records may also be deleted from a hash table.

Please
delete me.



Deleting a Record

- Records may also be deleted from a hash table.
- But the location must not be left as an ordinary "empty spot" since that could interfere with searches.



Deleting a Record

- Records may also be deleted from a hash table.
- But the location must not be left as an ordinary "empty spot" since that could interfere with searches.
- The location must be marked in some special way so that a search can tell that the spot used to have something in it.





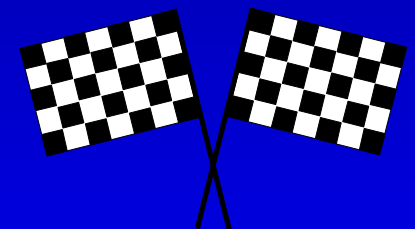
Summary

- ❑ Hash tables store a collection of records with keys.
- ❑ The location of a record depends on the hash value of the record's key.
- ❑ When a collision occurs, the next available location is used.
- ❑ Searching for a particular key is generally quick.
- ❑ When an item is deleted, the location must be marked in a special way, so that the searches know that the spot used to be used.

Presentation copyright 2010 Addison Wesley Longman,
For use with *Data Structures and Other Objects Using C++*
by Michael Main and Walter Savitch.

Some artwork in the presentation is used with permission from Presentation Task Force (copyright New Vision Technologies Inc) and Corel Gallery Clipart Catalog (copyright Corel Corporation, 3G Graphics Inc, Archive Arts, Cartesia Software, Image Club Graphics Inc, One Mile Up Inc, TechPool Studios, Totem Graphics Inc).

Students and instructors who use *Data Structures and Other Objects Using C++* are welcome to use this presentation however they see fit, so long as this copyright notice remains intact.



THE END