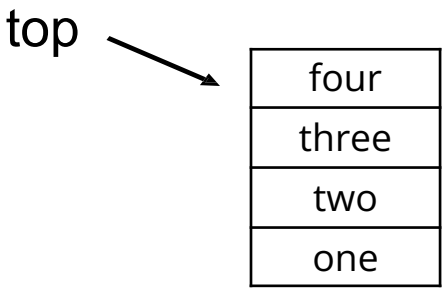
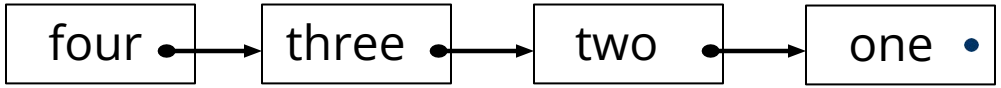
Stack Implementation
ArrayStack

COMP128 Data Structures

Implementations of a stack



Array Implementation



Linked Node Implementation

Abstract Data Type and Data Structures

	Conceptually	Implementations	
	Abstract Data Type	**Data Structure**	**Data Structure**
Java Collection Framework (use directly)	**Deque** Interface	**ArrayDeque** Class	**LinkedList** Class
Java Foundations Textbook (learn how to implement)	**StackADT** Interface	**ArrayStack** Class	**LinkedStack** Class

Abstract Data Type and Data Structures

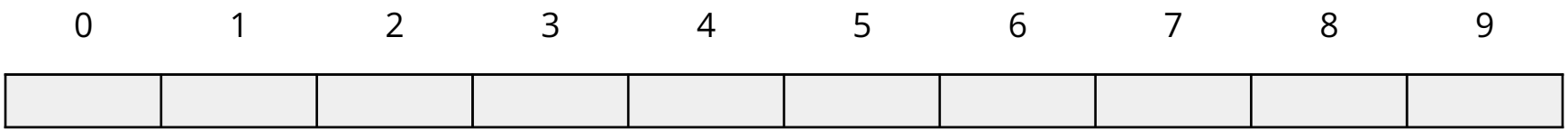
	Conceptually	Implementations	
	Abstract Data Type	**Data Structure**	**Data Structure**
Java Collection Framework (use directly)	**Deque** Interface	**ArrayDeque** Class	**LinkedList** Class
Java Foundations Textbook (learn how to implement)	**StackADT** Interface	**ArrayStack** Class	**LinkedStack** Class

Abstract Data Type and Data Structures

	Conceptually	Implementations	
	Abstract Data Type	**Data Structure**	**Data Structure**
Java Collection Framework (use directly)	**Deque** Interface	**ArrayDeque** Class	**LinkedList** Class
Java Foundations Textbook (learn how to implement)	**StackADT** Interface	**ArrayStack** Class	**LinkedStack** Class

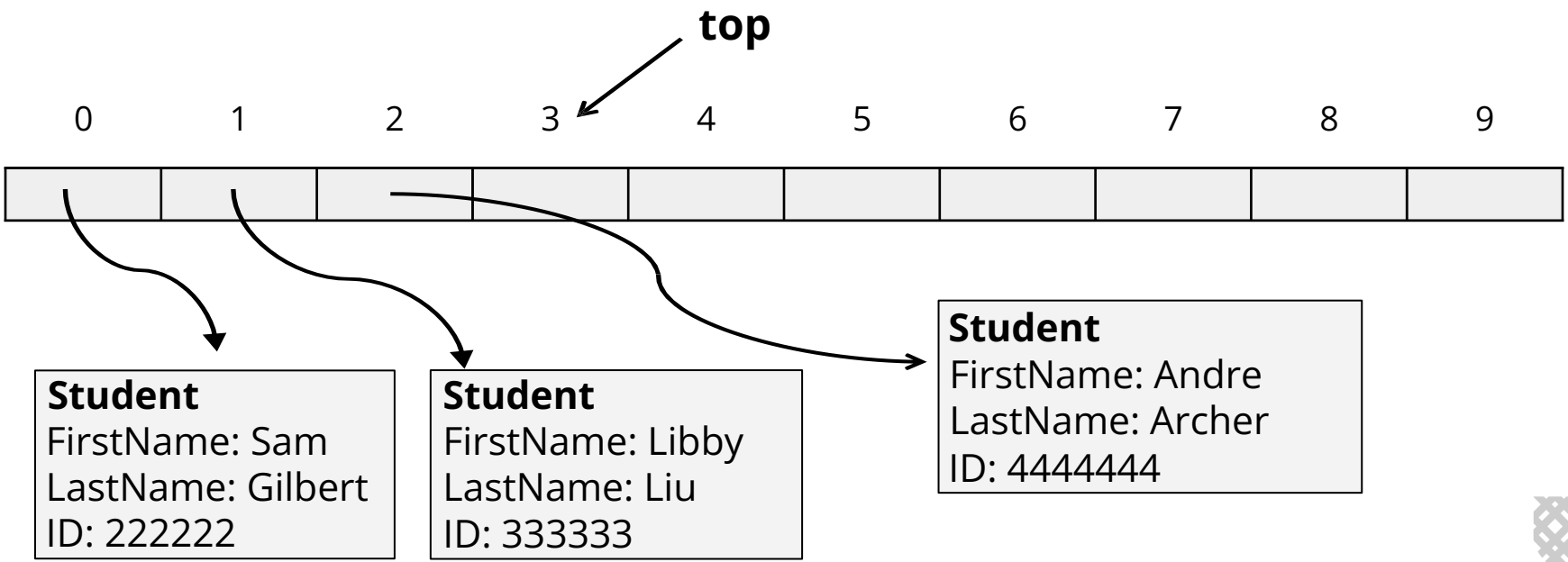
Implementation with an array

The array implementation of a stack has:



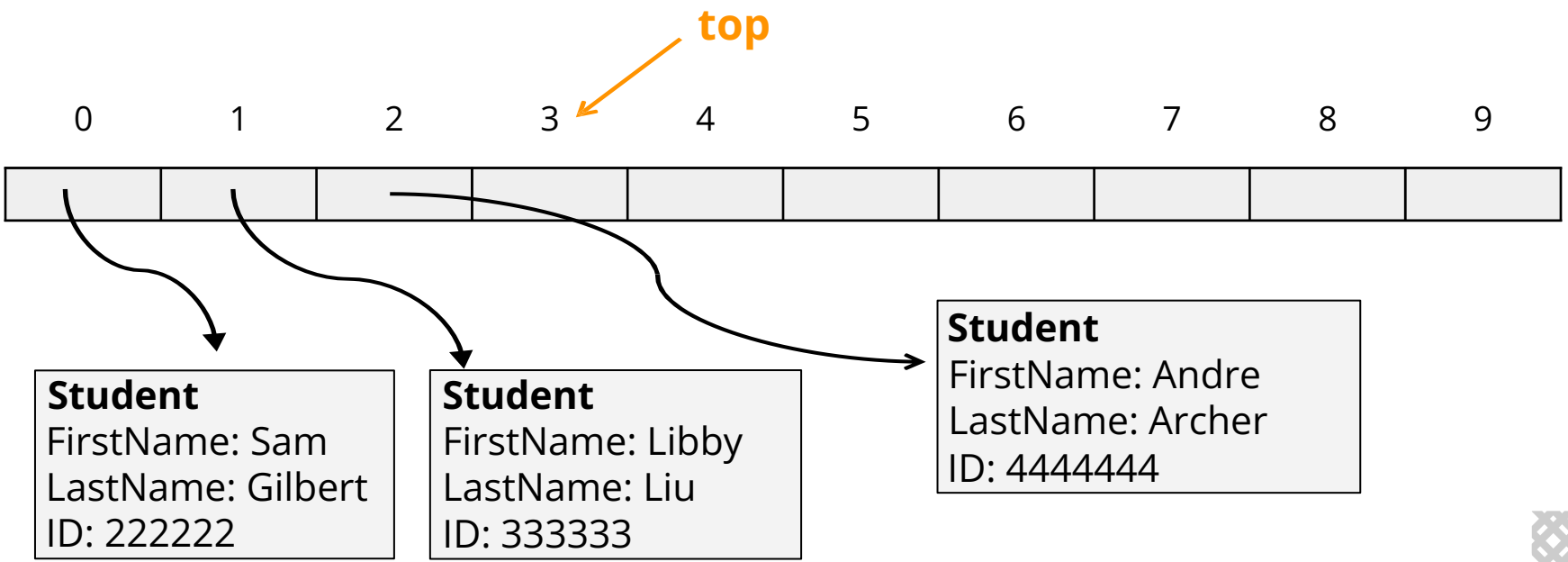
The array implementation of a stack has:

An array that contains references to Objects, such as instances of type Student An int that keeps track of the top of the stack



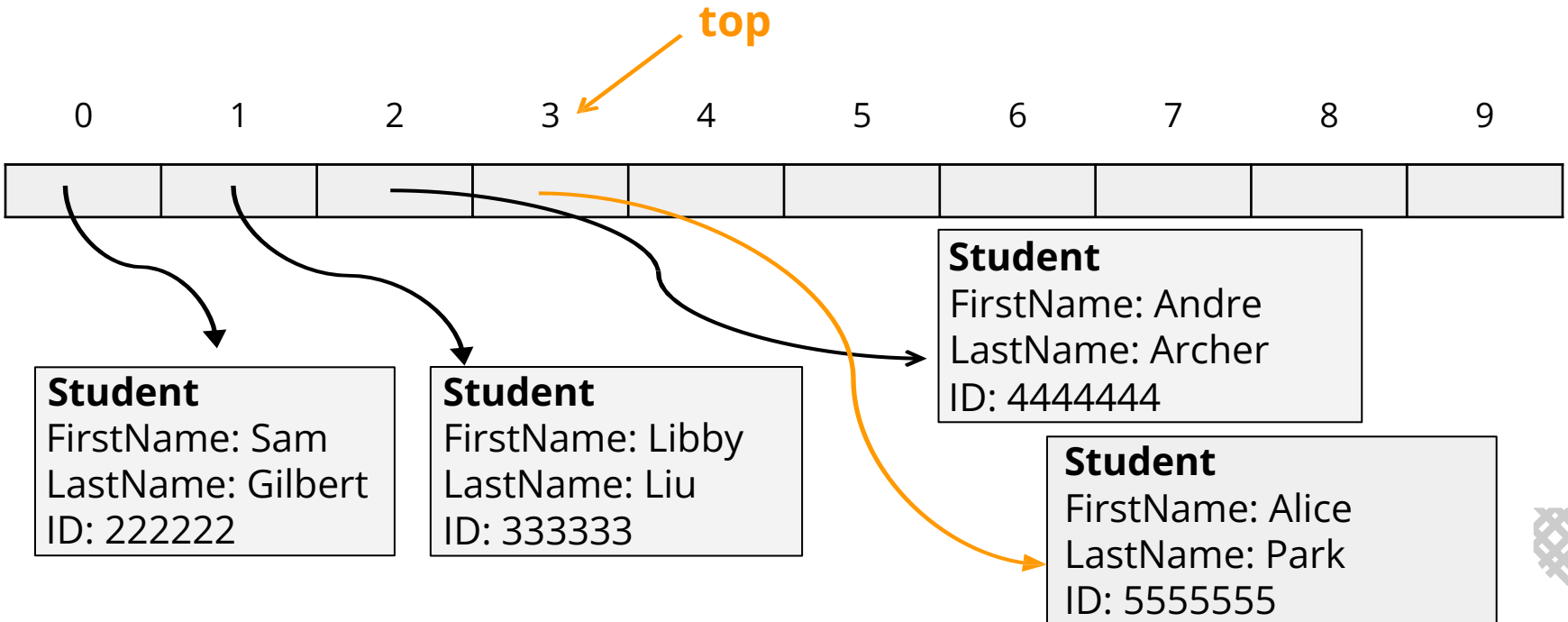
The array implementation of a stack has:

1. **Check if there is space and expand if necessary**



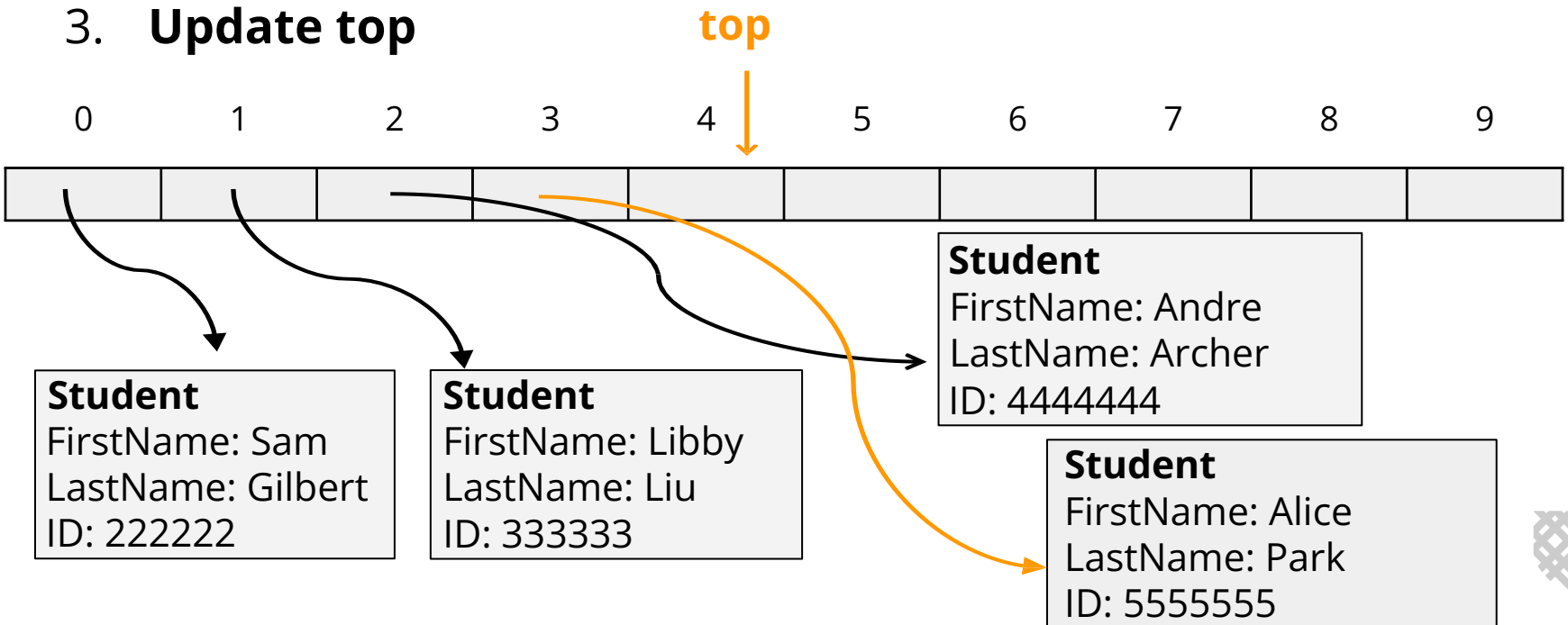
The array implementation of a stack has:

1. Check if there is space and expand if necessary
2. **Insert the object at the top position**



The array implementation of a stack has:

1. Check if there is space and expand if necessary
2. Insert the object at the top position
3. **Update top**



Expanding the Array

Expanding the Array

If the array is full and an object is pushed on the stack, you should:

- Create a new array that is twice the size of the current one.
- Copy all the old items into the new array
- Add the new one as usual

Time Complexity

Method	Array Stack	Linked Stack
push(T elem)		
pop()		
peek()		
isEmpty()		

Time Complexity

Method	Array Stack	Linked Stack
push(T elem)	Amortized O(1)	
pop()	O(1)	
peek()	O(1)	
isEmpty()	O(1)	

In-class Activity

ArrayStack Implementation