Data Structures

Arthur Hoskey, Ph.D. Farmingdale State College Computer Systems Department



- Describe a stack and its operations at a logical level
- Demonstrate the effect of stack operations using a particular implementation of a stack
- Implement the Stack ADT, using an arraybased implementation





Stacks of Coins and Bills



•What do these composite objects all have in common?



Stack

An abstract data type in which elements are added and removed from only one end (LIFO)



Stack

What operations would be appropriate for a stack?



9

Here is the interface for the Stack ADT:

public interface Stack {
 void push(int item) throws Exception;
 void pop() throws Exception;
 int top() throws Exception;
 void makeEmpty();

boolean isEmpty();
boolean isFull();

The public interface of a stack should be the same for both the array-based and linked implementations



Array-based **private** members

class StackArrayBased implements Stack
 Declare int maxItems
 Declare int top
 Declare int items[];

// Public members go here...
}
[0] [1] [2] [M..]
stack
.items
.top

Stack (Array)

 What does a stack look like if we insert the following elements (in the given order): 11, 14, 32



- Insert: 11, 14, 32
 Which stack was created from the above
 - insertions?









 What does a stack look like before any elements are added.



Here is a stack with 0 elements.

• Note: Why does top contain the value -1?



Stack (Array)

 Now run the following: *Declare StackArrayBased s Set s to new StackArrayBased instance s.push(11) What happens???*



Stack (Array) - push

s.push(11)

- Top has the value 0
- The value 11 is in element 0 of the array.



Stack (Array) - push

What if we ran the following code:
 s.push(14) s.push(32) What happens???



Stack (Array) - push

What if we ran the following code:
 s.push(14) s.push(32)

Stack grows in this direction



Stack (Array) - push

 What code would we run to remove the top element of the stack?



Stack (Array)





What if we run the following code on the stack below:

s.push(33)

Stack grows in this direction



Stack (Array) - pop

s.push(33) THIS CAUSES A "Stack Overflow".

Stack grows in this direction Stack 11 14 32 50 83 77 91 55 22 66 0 1 2 3 4 5 6 7 8 9 top 9

Stack (Array) - pop

Stack overflow

•The condition that results from trying to **push** an element on to a full stack

Stack (Array) - Overflow

What if we run the following code on the stack below:

s.pop()



Stack (Array) - pop

s.pop() THIS CAUSES A "Stack Underflow".



Stack (Array) - Underflow

Stack underflow The condition that results from trying to pop an empty stack

Stack (Array) - Underflow

Stack (Array)

push(int item) throws Exception

if (stack is full)

throw exception "Full Stack"

Increment top

Set items[top] to item

Stack (Array) - push

pop() throws Exception

if (stack is empty)

throw exception "Empty Stack"

Decrement top

Stack (Array) - pop

makeEmpty() Set top to -1

Deletes <u>ALL</u> of the elements in the stack.

Stack (Array) – makeEmpty

- To get data from the stack you need to call the top() method (for this particular ADT).
- Note: There are other Stack ADTs where the pop() method actually returns the value of the top item.



top() returns int throws Exception

if (stack is empty)

throw exception "Empty Stack"

return items[top]

Stack (Array) - top

 $\ensuremath{\textcircled{}}$ 2022 Arthur Hoskey. All rights reserved.

push, pop and top ALL throw exceptions.

 Since these methods throw exceptions you MUST call them inside of a try/catch block.

For example...





Stack (Array)

Operation	Cost
isFull	O(1)
isEmpty	O(1)
push	O(1)
рор	O(1)
Constructor	O(1)

Stack (Array) – Big O



End of Slides