

Stack (Data Structure)

Prof. Wade Fagen-Ulmschneider

I ILLINOIS

ALMA MATER
DEDICATED TO THIS DAY CHILDREN
FOR THE FUTURE

A stack is a **last-in first-out** data structure that is similar to a pile (“stack”) of papers.



Stack ADT

create → Creates an empty stack

push → Adds data to the top of the stack

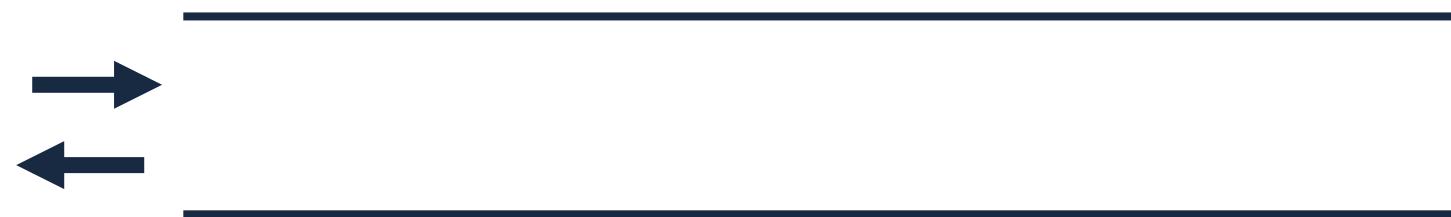
pop → Removes data from the top of the stack

empty → Returns true if the stack is empty

Example:

Consider four operations:

- Create a stack of integers.
- Push the number 4 to the stack.
- Push the number 2 to the stack.
- Pop a number from the stack.



stack/main.cpp

```
8 #include <iostream>
9 #include <stack>
10
11 int main() {
12     // Create a std::stack:
13     std::stack<std::string> s;
14
15     // Add several strings to the stack:
16-8     s.push( "Orange" );    s.push( "Blue" );    s.push( "Illinois" );
19
20     // Print the front of the stack out and pop it off:
21     std::cout << "First pop(): " << s.top() << std::endl;
22     s.pop();
23
24     // Add another string and then print ouf the front of the stack:
25     s.push( "Illini" );
26     std::cout << "Second pop(): " << s.top() << std::endl;
27
28     return 0;
29 }
```

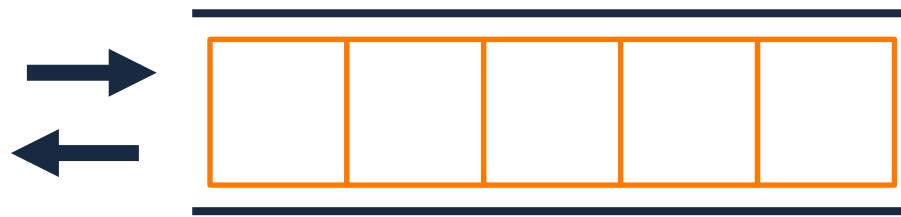
Implementation

Array-based:



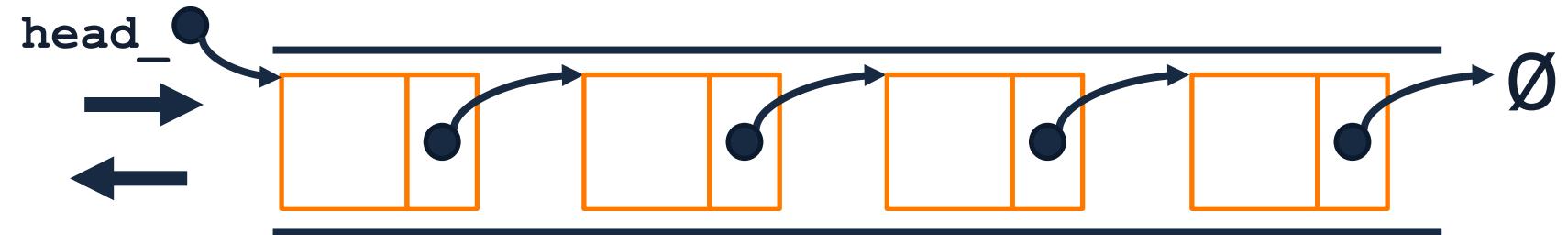
Example:

```
s.push(1);
s.push(2);
s.push(3);
s.pop();
s.push(4);
s.push(5);
s.pop();
s.push(6);
s.push(7);
s.push(8);
```



Implementation

List-based:



Objective: Create a Queue Data Structure

	Array	Linked List (Singly-linked list is sufficient)
Create	O(1)	O(1)
Push	O(1)* <small>*Amortized runtime; occasional need to double the capacity of the array.</small>	O(1)
Pop	O(1)* <small>*Amortized runtime; occasional need shrink the size of the array to free unused memory.</small>	O(1)
Empty	O(1)	O(1)

Stack

- A stack is a last-in first-out (LIFO) data structure that is similar to a pile (“stack”) of papers.
- A stack may be implemented with an array or a linked list.
 - Both an array-based and a list-based implementation can be built to run in constant, $O(1)$ running time.

