

In class Queue as 2 Stacks - up to 4 in a group

Stack (LIFO):

Foo pop() - returns the top element on the stack

void push(Foo) - adds the parameter to the top of the stack

boolean isEmpty() - returns true if the stack is empty (surprise!)

Queue (FIFO):

Foo dequeue() - returns the first thing on the queue

void enqueue(Foo) - adds the parameter to the end of the queue

boolean isEmpty() - um... similar to isEmpty for Stack

Task: Write a Queue class for elements of type Foo. The only data structures you may use are two Stacks. Send someone up to tell me as soon as you are done. Write your methods on the back and hand in. Extra Credit! Implement a queue with only *one* Stack.

Group members:

-
-
-
-

In class Queue as 2 Stacks - up to 4 in a group

Stack (LIFO):

Foo pop() - returns the top element on the stack

void push(Foo) - adds the parameter to the top of the stack

boolean isEmpty() - returns true if the stack is empty (surprise!)

Queue (FIFO):

Foo dequeue() - returns the first thing on the queue

void enqueue(Foo) - adds the parameter to the end of the queue

boolean isEmpty() - um... similar to isEmpty for Stack

Task: Write a Queue class for elements of type Foo. The only data structures you may use are two Stacks. Send someone up to tell me as soon as you are done. Write your methods on the back and hand in. Extra Credit! Implement a queue with only *one* Stack.

Group members:

-
-
-
-