

# CSC148 – Introduction to Computer Science

Lecture 2: Abstract Data Types – Stacks  
and Queues

Sean Henderson

# What's an ADT?

- Defining a data type without reference to how it is to be implemented
- For example, the List
- You know operations in Python you can perform in it (append, pop, len, etc.) but this says nothing about how the list is implemented
- Hides these details from the user
- Encapsulation!

# What's an ADT?

- Other examples?

# Stack

- Simple data structure for keeping track of a sequence of objects
- Objects are added (pushed) and removed (popped) from the top
- Removed in the opposite order than in which they were inserted
- Last-In-First-Out (LIFO)

# Practical Stack Examples

- Keeping track of function calls
- Check that parenthesis are balanced in an expression
- The undo function
- In compilers, to parse the source code of your programs

# Stack ADT

- `push(obj)` – Add `obj` to the top of the stack
- `pop()` – Remove and return the item on the top of the stack
- `peek()` – Return the item on the top of the stack without removing it
- `empty()` – Return true iff there are no items on the stack
- `size()` – How many items are currently on the stack

Wing-101

# Queue

- Simple data structure function much like you would expect a line to
- Items are removed in the same order they were inserted
- First-In-First-Out (FIFO)

# Practical Queue Examples

- Print jobs (sort of...)
- Operating systems deciding which processes to run (sort of...)
- Graphical User Interfaces (GUIs) to keep track of events (like mouse clicks, key presses, etc.)

# Queue ADT

- `enqueue(obj)` – Add `obj` to the back of the queue
- `dequeue()` – Remove and return the item on the front of the queue
- `front()` – Return the item on the front of the queue without removing it
- `empty()` – Return `true` iff there are no items in the queue
- `size()` – How many items are currently in the queue

Wing-101