CSCI 136:
Data Structures
and
Advanced Programming

Lecture 20

Ordered Structures

Instructor: Dan Barowy

Williams

Announcements

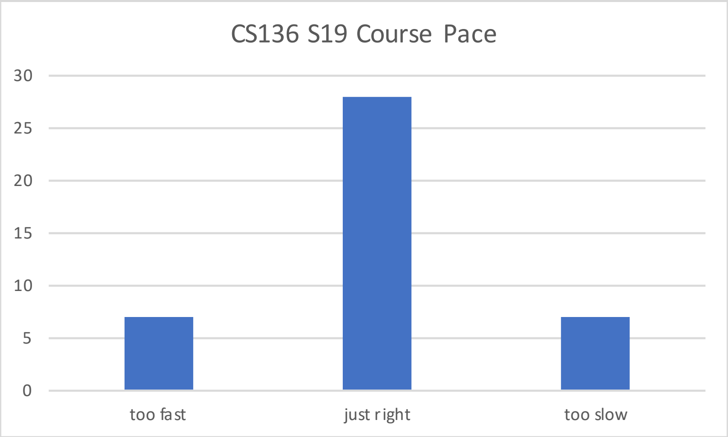
No Barowy office hours this Friday

PRE-LAB 0: due tonight

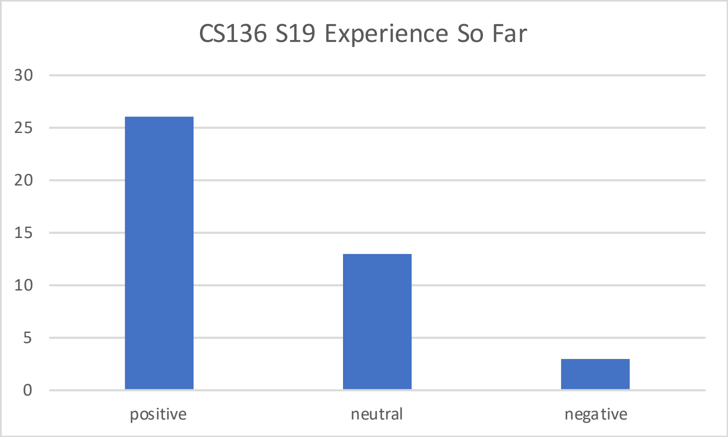
PRE-LAB 1: due in lab on Wed

One-on-one meetings

Feedback: thank you



If you're in the tails, I am happy to talk.



Ambivalent/negative? Happy to discuss.

Least popular features

Pop quizzes

Random partners

Purpose of quizzes

To incentivize reading before class.

Purpose of partners

To prepare you to work someplace not imaginary.

Outline

1. Ordered structures (cont'd)
2. Infix to postfix algorithm

Quiz

Ordered structures

Recall

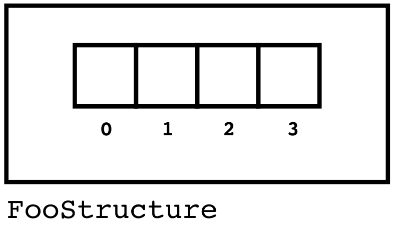
We want a **data structure** that has the same API as other structures (e.g., Vector, List, etc.) but that also **keeps itself ordered** all the time.

This is what we call an **ordered structure**.

Should be possible

Simple inductive proof.

Start with an **empty data structure**.

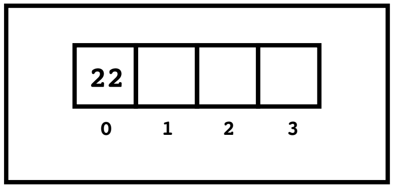


Is FooStructure ordered? **Yes.**

Should be possible

Simple inductive proof.

Add an element (any element). **Insert it in order.**



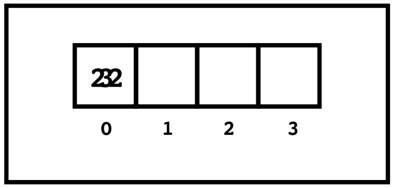
FooStructure

Is FooStructure ordered? **Yes.**

Should be possible

Simple inductive proof.

Add another element (any element). **Insert it in order.**



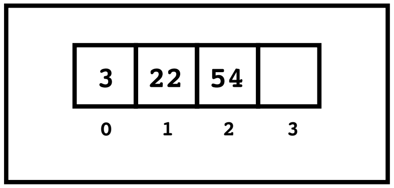
FooStructure

Is FooStructure ordered? **Yes.**

Should be possible

Simple inductive proof.

Add another element (any element). **Insert it in order.**



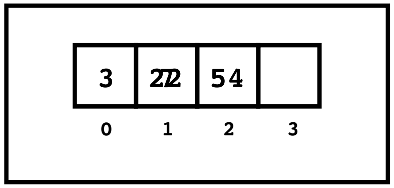
FooStructure

Is FooStructure ordered? **Yes.**

Should be possible

Simple inductive proof.

Add another element (any element). **Insert it in order.**



FooStructure

Is FooStructure ordered? **Yes.**

OrderedStructure

Recall: we will implement an OrderedStructure to **signal our intent** that the structure should always be ordered.

Recall: its elements must have a **special property**: they are `Comparable<T>`.

OrderedVector

Let's resume implementing an OrderedVector.

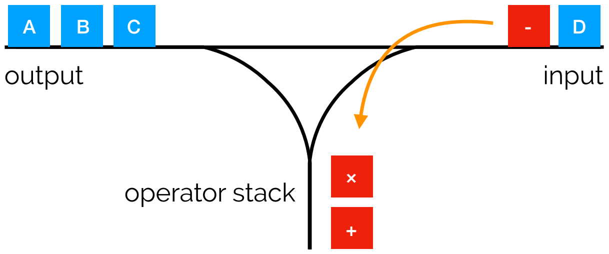
(code)

OrderedVector

What is the biggest limitation of an OrderedVector?

How might you overcome that limitation?

Shunting yard algorithm



Shunting yard algorithm

Converts **infix** expressions to **postfix** expressions.

Recall the hand-wavy method I gave before.

This one is better.

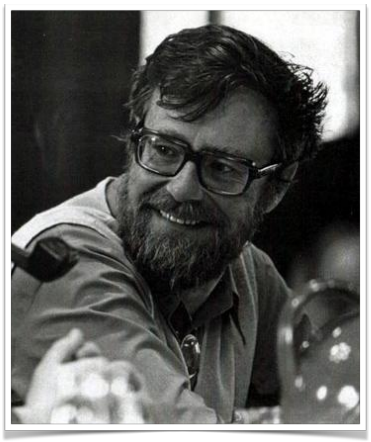
Utilizes a **stack** and a **queue**.

Shunting yard algorithm

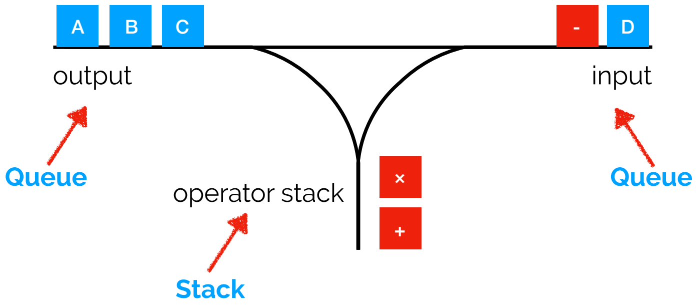
Shunting yard algorithm

Invented by Edsger Dijkstra in 1961 while he was working on the ALGOL programming language.

Won the Turing Award in 1972 for his work on ALGOL and many, many, many other things.



Shunting yard algorithm



Shunting yard algorithm

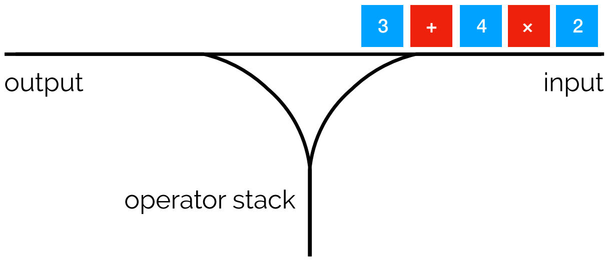
Pseudocode (slightly simplified):

```
while there are tokens to be read:
    read a token.
    if the token is a number, then:
        push it to the output queue.
    if the token is an op, then:
        while ((there is an operator at the top of the op stack with greater precedence) or
               (the op at the top of the op stack has equal precedence and is left associative)):
            pop ops from the op stack onto the output queue.
        push it onto the op stack.
if there are no more tokens to be read:
    while there are still op tokens on the stack:
        pop the op from the op stack onto the output queue.
exit.
```

Shunting yard algorithm

Example: 3 + 4 × 2

Example: 3 + 4 × 2

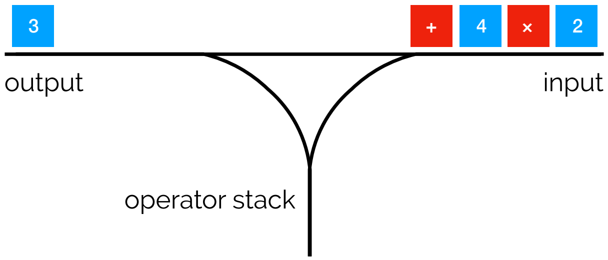


output

input

operator stack

Example: 3 + 4 × 2

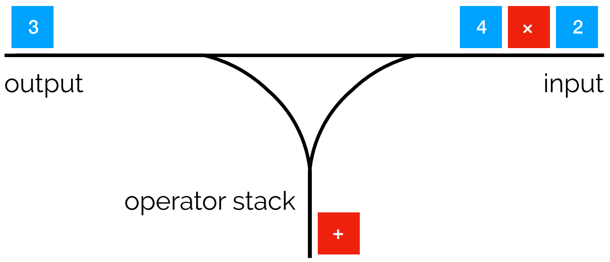


output

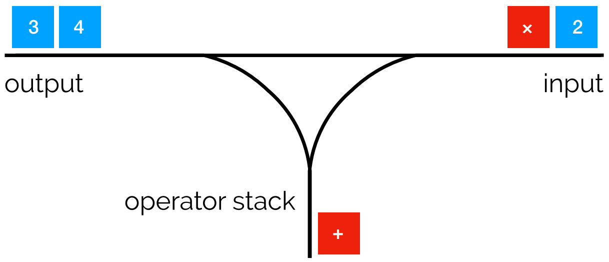
input

operator stack

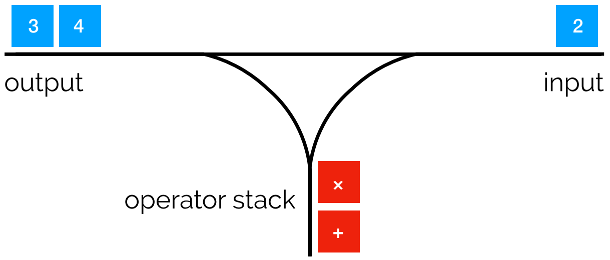
Example: 3 + 4 × 2



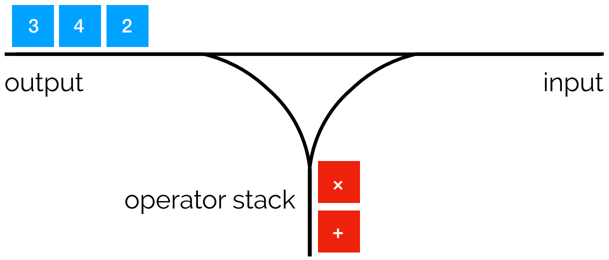
Example: 3 + 4 × 2

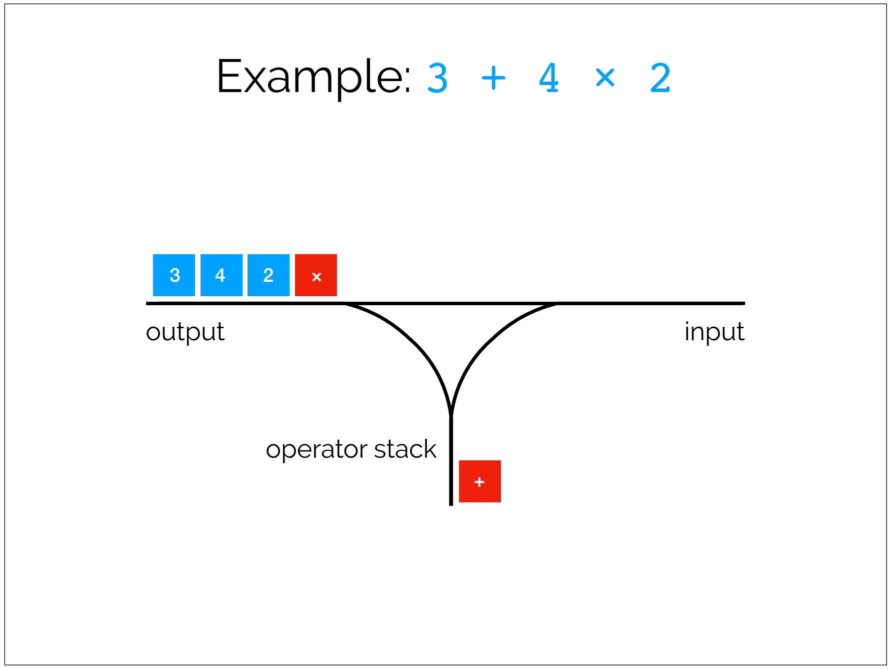


Example: 3 + 4 × 2

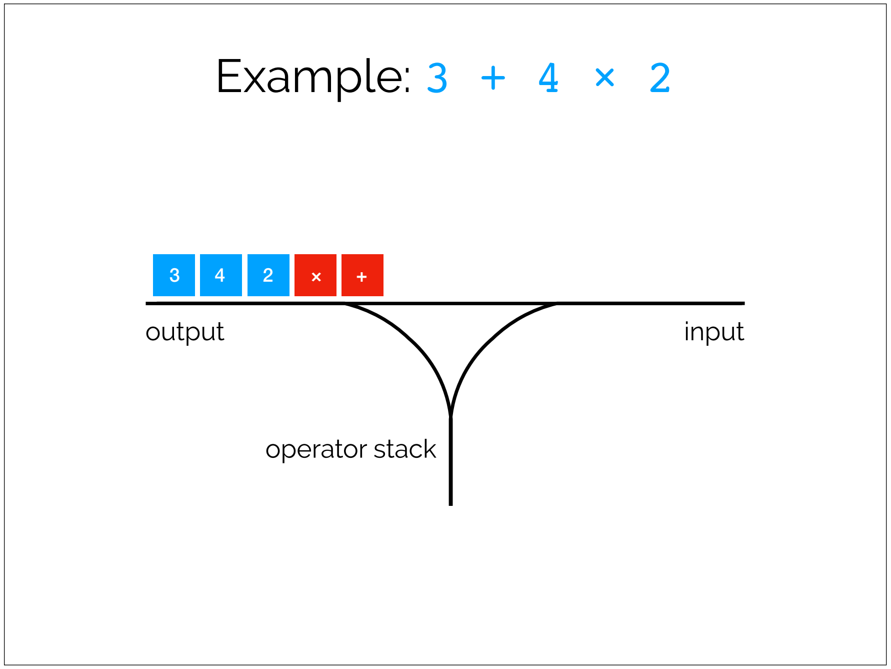


Example: 3 + 4 × 2

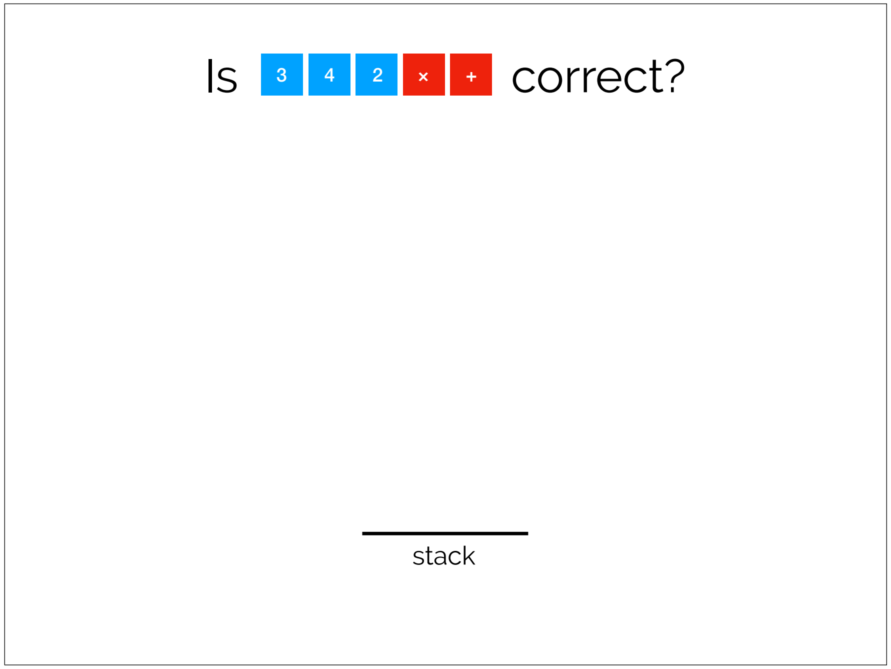




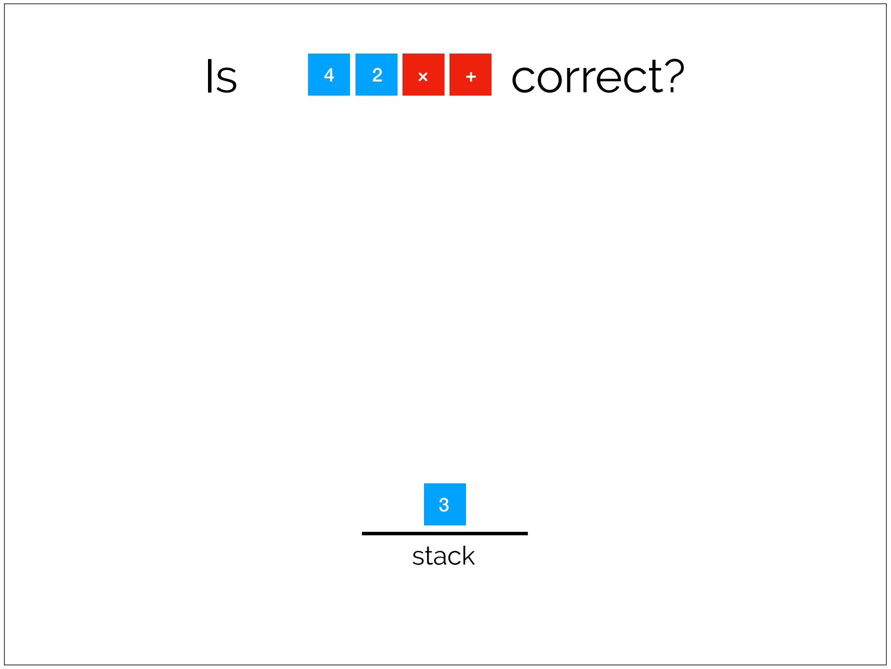
Example: 3 + 4 × 2



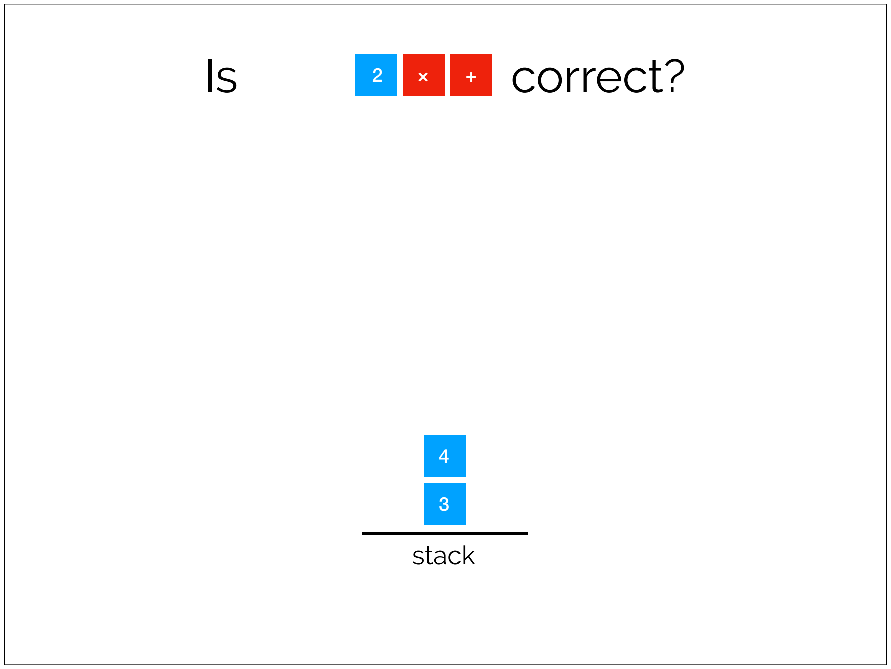
Example: 3 + 4 × 2



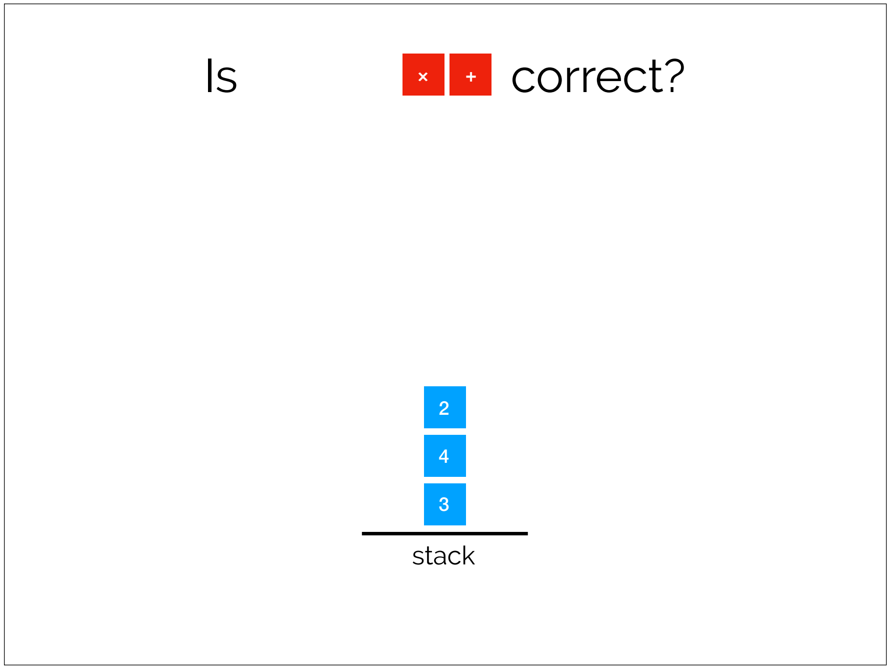
Is 3 4 2 × + correct?



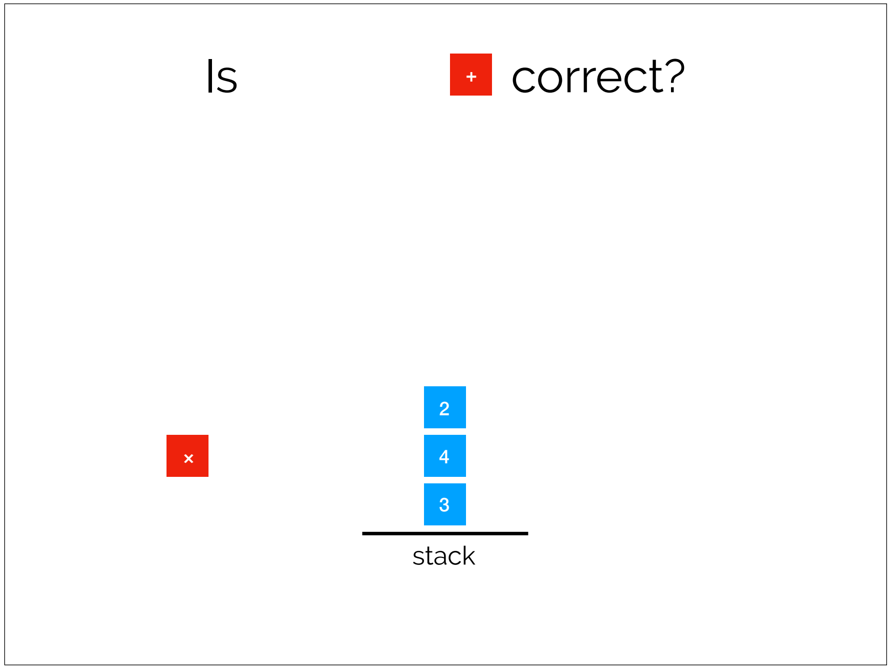
Is 4 2 × + correct?



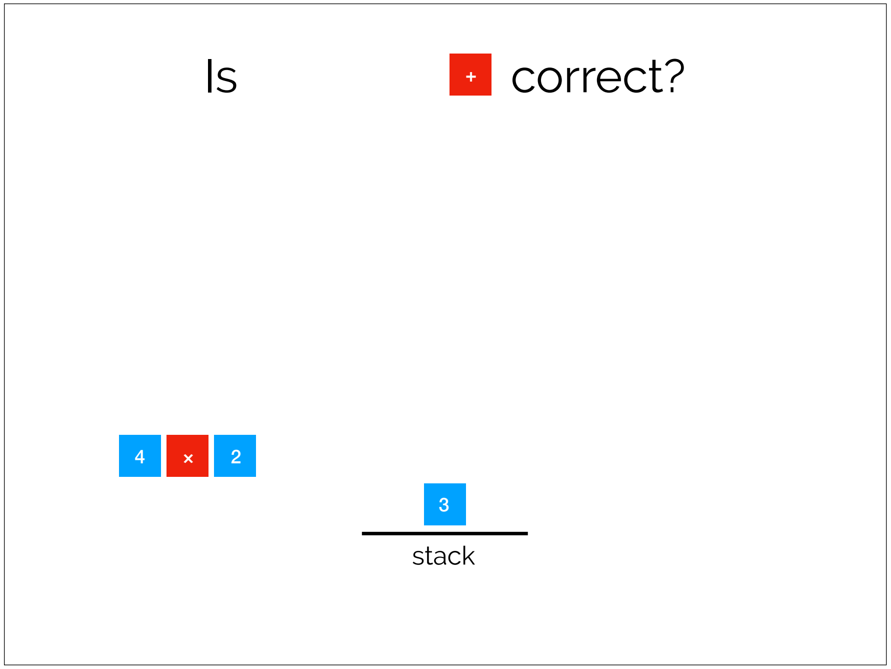
Is 2 × + correct?



Is × + correct?

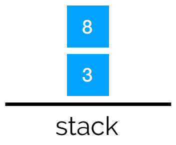


Is + correct?

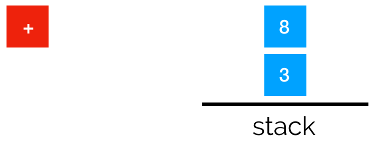


Is + correct?

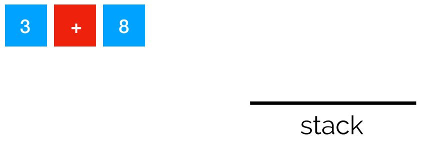
Is + correct?



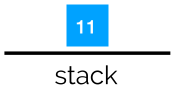
Is correct?

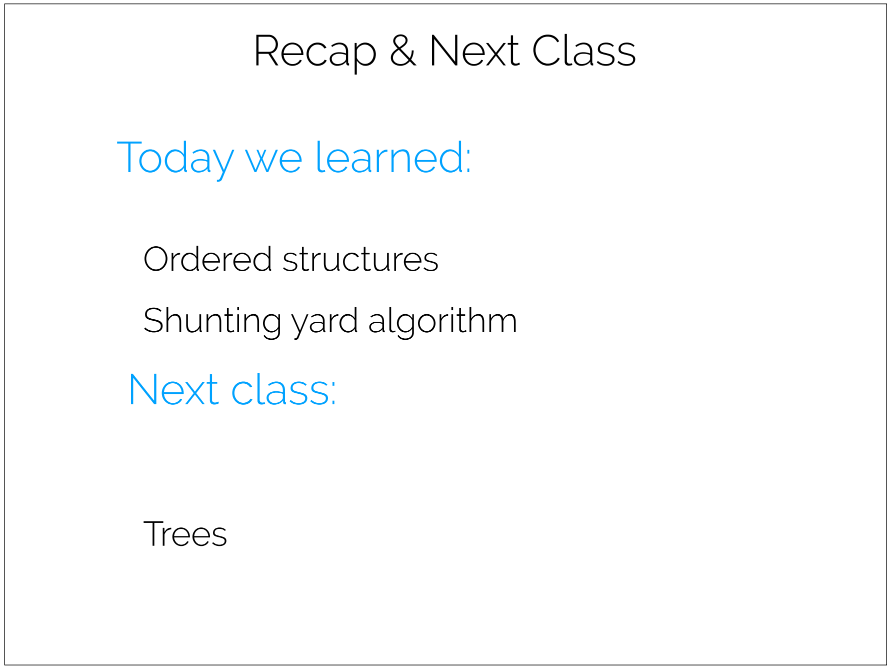


Is correct?



Is correct?





Recap & Next Class

Today we learned:

Ordered structures

Shunting yard algorithm

Next class:

Trees