Warmup

A programmer's wife tells him, "Would you mind going to the store and picking up a loaf of bread. Also, if they have eggs, get a dozen."

The programmer returns with 12 loaves of bread.

Section 3:
HW4, ADTs, and more

Slides by Alex Mariakakis

with material from Krysta Yousoufian,
Mike Ernst, Kellen Donohue

Agenda

- Announcements
    - HW3: due tonight
    - HW4: due next Thursday
- Polynomial arithmetic
- Abstract data types (ADT)
- Representation invariants (RI)

HW4: Polynomial Graphing Calculator

- **Problem 0:** Write pseudocode algorithms for polynomial operations
- **Problem 1:** Answer questions about RatNum
- **Problem 2:** Implement RatTerm
- **Problem 3:** Implement RatPoly
- **Problem 4:** Implement RatPolyStack
- **Problem 5:** Try out the calculator

Polynomial Addition

$$(5x^4 + 4x^3 - x^2 + 5) + (3x^5 - 2x^3 + x - 5)$$

Polynomial Addition

$(5x^4 + 4x^3 - x^2 + 5) + (3x^5 - 2x^3 + x - 5)$

$$\begin{array}{rrrrrrr} & & 5x^4 & + 4x^3 & - x^2 & & + 5 \\ + & 3x^5 & & - 2x^3 & & + x & - 5 \end{array}$$

Polynomial Addition

$$(5x^4 + 4x^3 - x^2 + 5) + (3x^5 - 2x^3 + x - 5)$$

$$\begin{array}{lllllll}
 & & 5x^4 & + 4x^3 & - x^2 & 0x & + 5 \\
+ & 3x^5 & 0x^4 & - 2x^3 & 0x^2 & + x & - 5
\end{array}$$

Polynomial Addition

$$(5x^4 + 4x^3 - x^2 + 5) + (3x^5 - 2x^3 + x - 5)$$

$$\begin{array}{ccccccc}
 & & 5x^4 & + 4x^3 & - x^2 & 0x & + 5 \\
+ & 3x^5 & 0x^4 & - 2x^3 & 0x^2 & + x & - 5 \\
\hline
 & 3x^5 & + 5x^4 & - 2x^3 & - x^2 & + x & + 0
\end{array}$$

Polynomial Subtraction

$(5x^4 + 4x^3 - x^2 + 5) + (3x^5 - 2x^3 + x - 5)$

$$\begin{array}{rrrrrrr} & & 5x^4 & + 4x^3 & - x^2 & & + 5 \\ - & 3x^5 & & - 2x^3 & & + x & - 5 \end{array}$$

Polynomial Subtraction

$$(5x^4 + 4x^3 - x^2 + 5) + (3x^5 - 2x^3 + x - 5)$$

$$\begin{array}{lcccccc}
 & & 5x^4 & + 4x^3 & - x^2 & 0x & + 5 \\
- & 3x^5 & 0x^4 & - 2x^3 & 0x^2 & + x & - 5
\end{array}$$

Polynomial Subtraction

$(5x^4 + 4x^3 - x^2 + 5) + (3x^5 - 2x^3 + x - 5)$

$$\begin{array}{rrrrrrr}
 & & 5x^4 & + 4x^3 & - x^2 & 0x & + 5 \\
- & 3x^5 & 0x^4 & - 2x^3 & 0x^2 & + x & - 5 \\
\hline
 & -3x^5 & + 5x^4 & + 6x^3 & - x^2 & - x & + 10
\end{array}$$

Polynomial Multiplication

$(4x^3 - x^2 + 5) * (x - 5)$

Polynomial Multiplication

$(4x^3 - x^2 + 5) * (x - 5)$

$$\begin{array}{lr} & 4x^3 - x^2 + 5 \\ * & x - 5 \\ \hline \end{array}$$

Polynomial Multiplication

$(4x^3 - x^2 + 5) * (x - 5)$

$$
\begin{array}{lrrrr}
 & 4x^3 & - x^2 & & + 5 \\
* & & & x & - 5 \\
\hline
 & -20x^3 & + 5x^2 & & - 25
\end{array}
$$

Polynomial Multiplication

$$(4x^3 - x^2 + 5) \; * \; (x - 5)$$

$$\begin{array}{llllll}
 & & & 4x^3 & -\ x^2 & +\ 5 \\
* & & & & x & -\ 5 \\
\hline
 & & -20x^3 & +\ 5x^2 & & -\ 25 \\
 & 4x^4 & -x^3 & & +\ 5x &
\end{array}$$

Polynomial Multiplication

$$(4x^3 - x^2 + 5) * (x - 5)$$

$$\begin{array}{lrrrrr}
 & & 4x^3 & -\ x^2 & & +\ 5 \\
* & & & & x & -\ 5 \\ \hline
 & & -20x^3 & +\ 5x^2 & & -\ 25 \\
+ & 4x^4 & -x^3 & & +\ 5x & \\ \hline
 & 4x^4 & -21x^3 & +\ 5x^2 & +\ 5x & -\ 25
\end{array}$$

Polynomial Division

$(5x^6 + 4x^4 - x^3 + 5) / (x^3 - 2x - 5)$

Polynomial Division

$$(5x^6 + 4x^4 - x^3 + 5) \,/\, (x^3 - 2x - 5)$$

$$x^3 - 2x - 5 \;\big|\overline{\;5x^6 + 4x^4 - x^3 + 5\;}$$

Polynomial Division

```
           ____________________________
1  0 -2 -5 |  5   0   4  -1   0   0   5
```

Polynomial Division

```
                              5
               _____________________________
1   0  -2  -5 |   5   0   4  -1   0   0   5
```

Polynomial Division

$$\begin{array}{r|rrrrrrr} & & & & 5 & & & \\ \cline{2-8} 1\quad 0\ \ -2\ \ -5 & 5 & 0 & 4 & -1 & 0 & 0 & 5 \\ & 5 & 0 & -10 & -25 & & & \end{array}$$

Polynomial Division

```
                        5
           _________________________
1  0 -2 -5 |  5  0  4  -1  0  0  5
              5  0-10 -25
            ----------------
              0  0 14  24
```

Polynomial Division

```
                          5
             ____________________________
1   0 -2 -5 |  5   0   4  -1   0   0   5
               5   0 -10 -25
               ______________
               0   0  14  24
                      14  24   0
```

Polynomial Division

```
                        5  0
            ┌───────────────────────
1   0 -2 -5 │ 5  0  4  -1  0  0  5
              5  0-10 -25
             ───────────────
              0  0 14  24
                   14  24 0
```

Polynomial Division

```
                       5  0
             ____________________
1  0 -2 -5 |  5  0  4 -1  0  0  5
              5  0-10 -25
             ____________
              0  0 14 24
                   14 24 0
                   14 24 0  0
```

Polynomial Division

```
                        5  0 14
           _________________________
1  0 -2 -5 |  5  0  4 -1  0  0  5
              5  0-10 -25
              --------------
              0  0 14  24
                   14  24 0
                   14  24 0  0
```

Polynomial Division

```
                       5  0 14
          ________________________
1  0 -2 -5|  5  0  4  -1  0  0  5
             5  0-10 -25
            _______________
             0  0 14  24
                  14  24 0
                  14  24 0  0
                  14  0 -28 -70
```

Polynomial Division

```
                        5  0 14
            ______________________
1  0 -2 -5  |  5  0  4  -1   0  0  5
               5  0-10 -25
               ____________
               0  0 14  24
                    14  24 0
                    14  24 0  0
                    14 0  -28 -70
                    _______________
                    0  24  28  70
```

Polynomial Division

```
                        5  0 14
            ________________________
1  0 -2 -5 |  5  0  4  -1  0  0  5
              5  0-10 -25
            ________________
              0  0 14  24
                   14  24 0
                   14  24 0  0
                   14  0 -28 -70
                 ____________________
                   0  24  28  70
                      24  28  70 5
```

Polynomial Division

```
                        5  0 14 24
1  0 -2 -5    5  0  4  -1  0  0  5
              5  0-10 -25
              0  0 14  24
                   14  24 0
                   14  24 0  0
                   14  0 -28 -70
                   0  24  28  70
                      24  28  70 5
                      24  0  -48 120
```

Polynomial Division

```
                        5  0 14 24
1  0 -2 -5    5  0  4  -1  0  0  5
              5  0-10 -25
              0  0 14  24
                   14  24 0
                   14  24 0  0
                   14  0 -28 -70
                   0  24  28  70
                      24  28  70 5
                      24  0  -48 120
                      0   28 118 125
```

Polynomial Division

$(5x^6 + 4x^4 - x^3 + 5)$ / $(x^3 - 2x - 5)$

$5x^3 + 14x + 24$

Polynomial Division

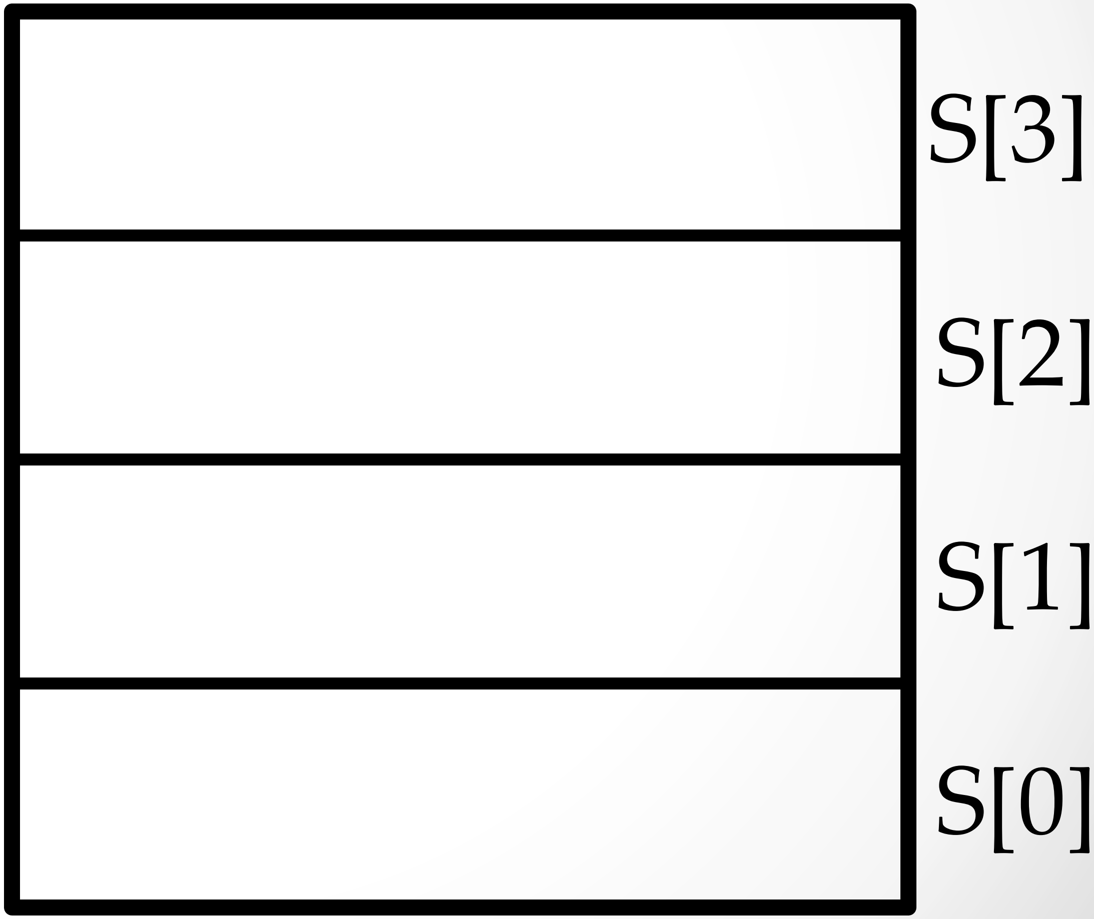
$(5x^6 + 4x^4 - x^3 + 5) / (x^3 - 2x - 5)$

$$5x^3 + 14x + 24 + \frac{28x^2 + 118x + 125}{x^3 - 2x - 5}$$

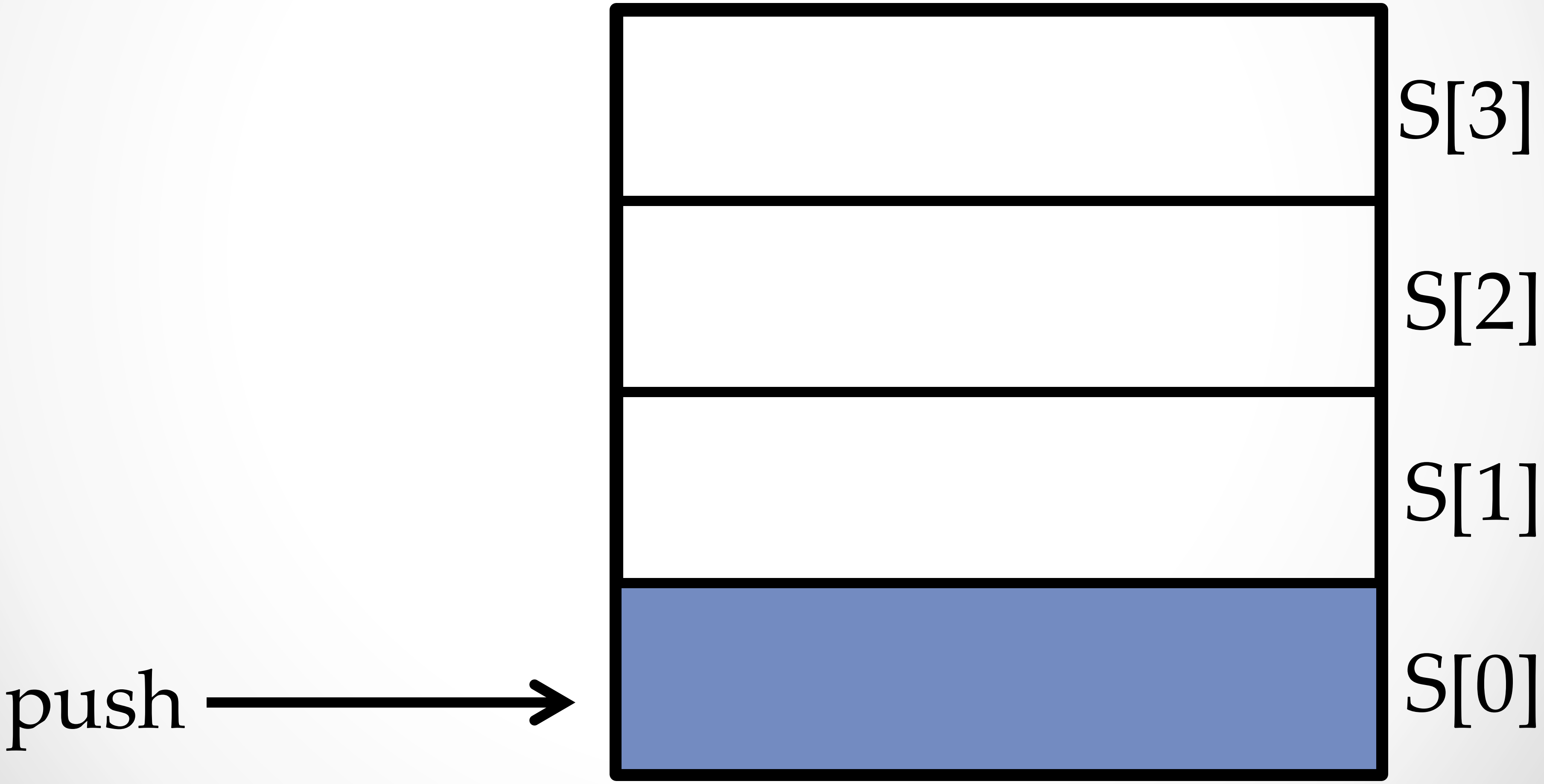
RatThings

- RatNum
  - ADT for a Rational Number
  - Has NaN
- RatTerm
  - Single polynomial term
  - Coefficient (RatNum) & degree
- RatPoly
  - Sum of RatTerms
- RatPolyStack
  - Ordered collection of RatPolys

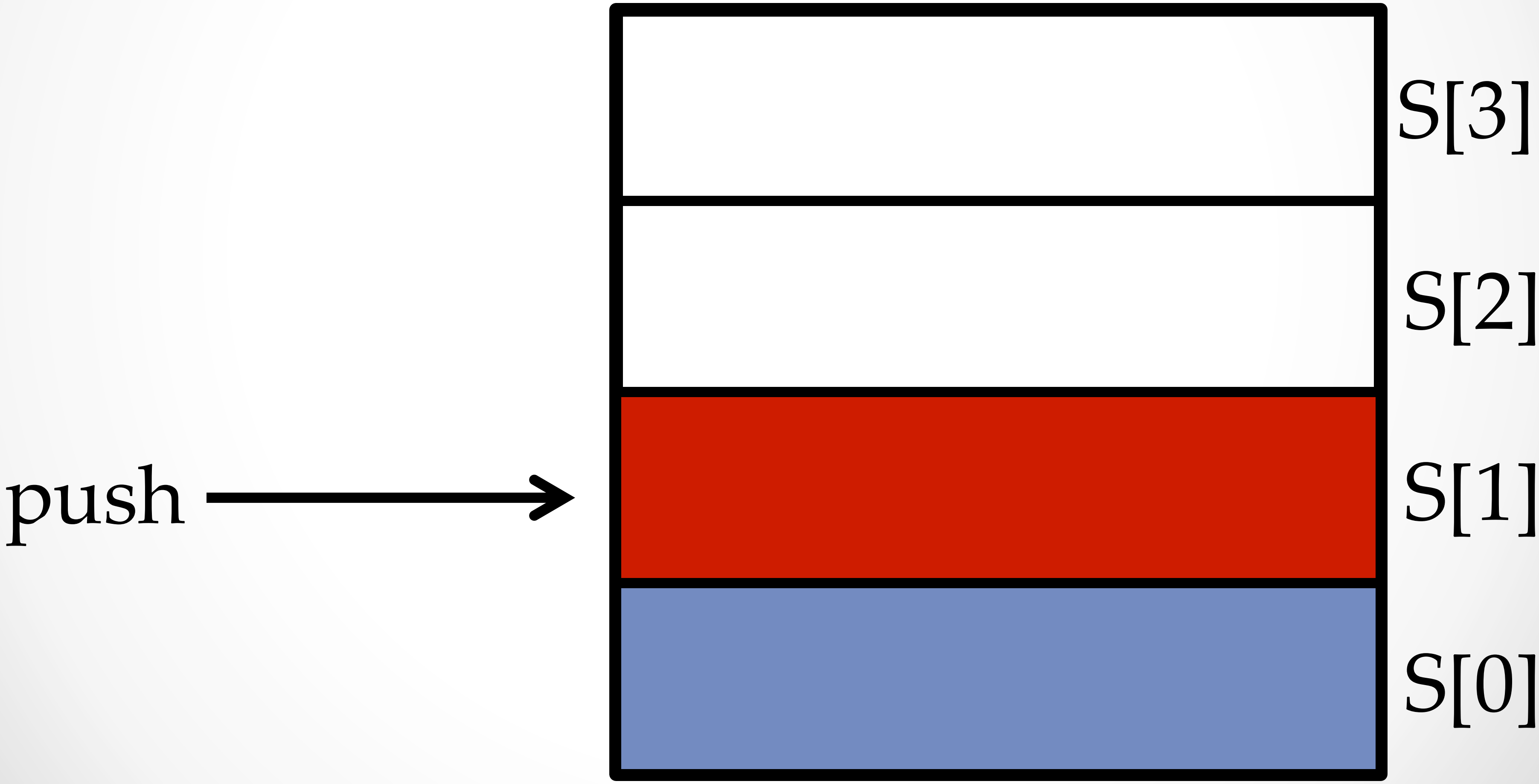
Stacks



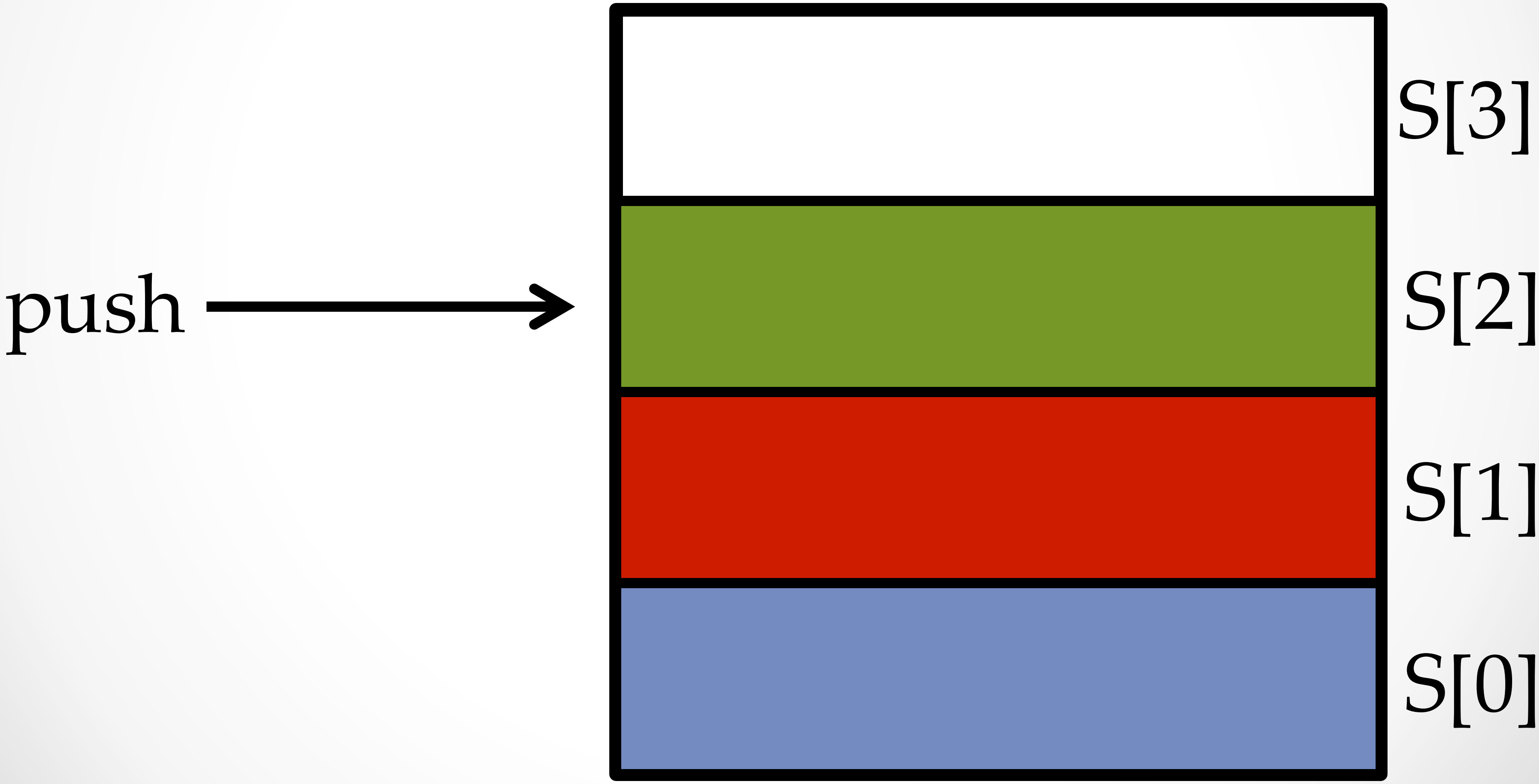
Stacks



Stacks

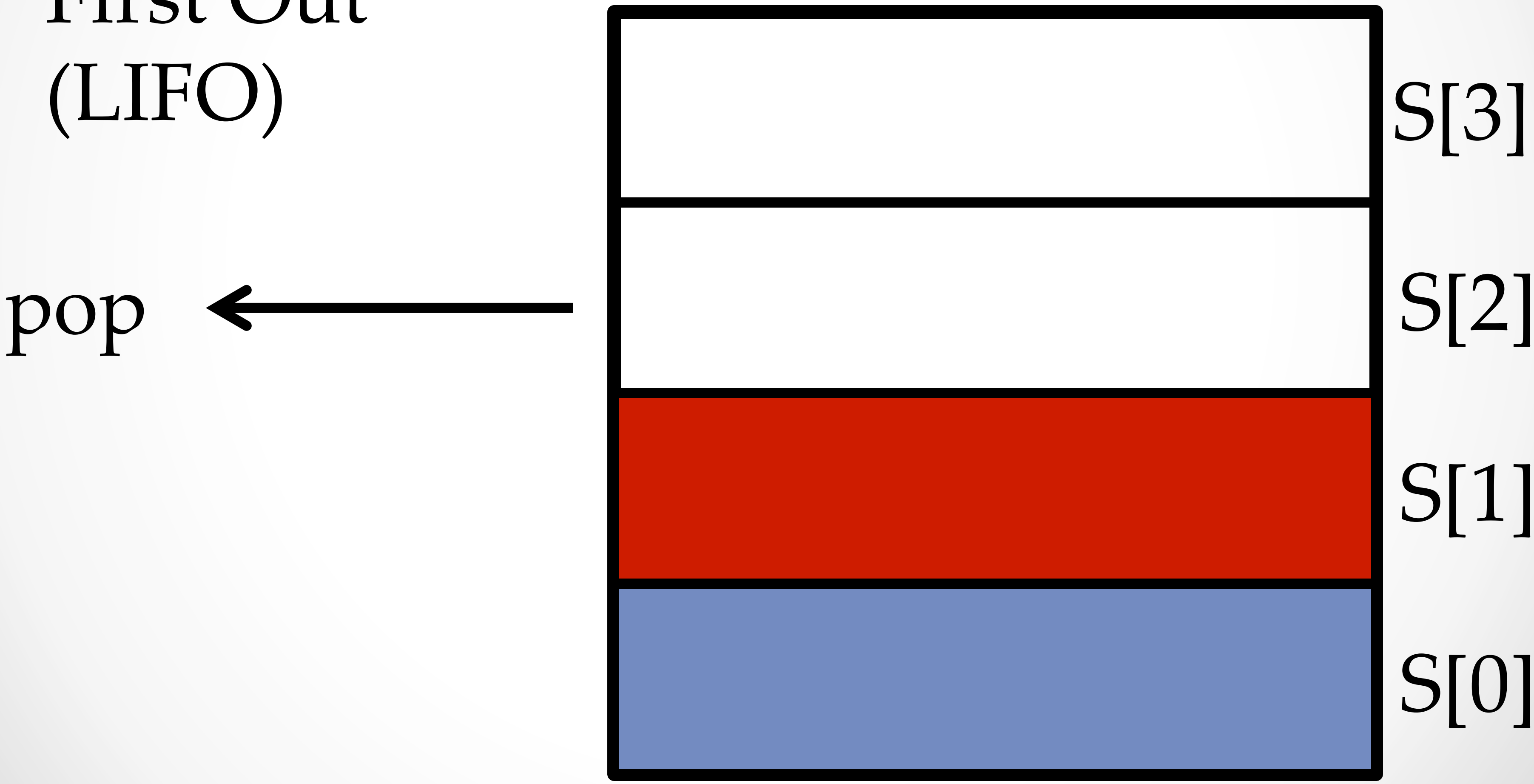


Stacks



Stacks

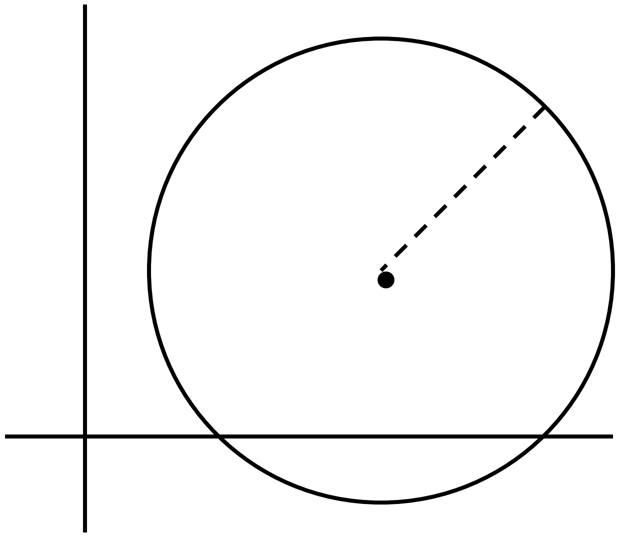
Last In,
First Out
(LIFO)



CalculatorFrame Demo

ADT Example: Circle

- Circle on the Cartesian coordinate plane



Abstraction Function

- Abstraction function: a **mapping** from **internal state** to **abstract value**
- Abstract fields may not map directly to representation fields
    - Circle has **radius** but not necessarily

      ```
      private int radius;
      ```
- Internal representation can be anything as long as it somehow encodes the abstract value

Representation Invariants

- Constrains an object’s internal state
- Defines what must be true for abstraction function to hold
- If representation invariant is violated, the object is “broken” – doesn’t map to any abstract value

Circle: Class Specification

What are the abstract fields (what the client sees)?

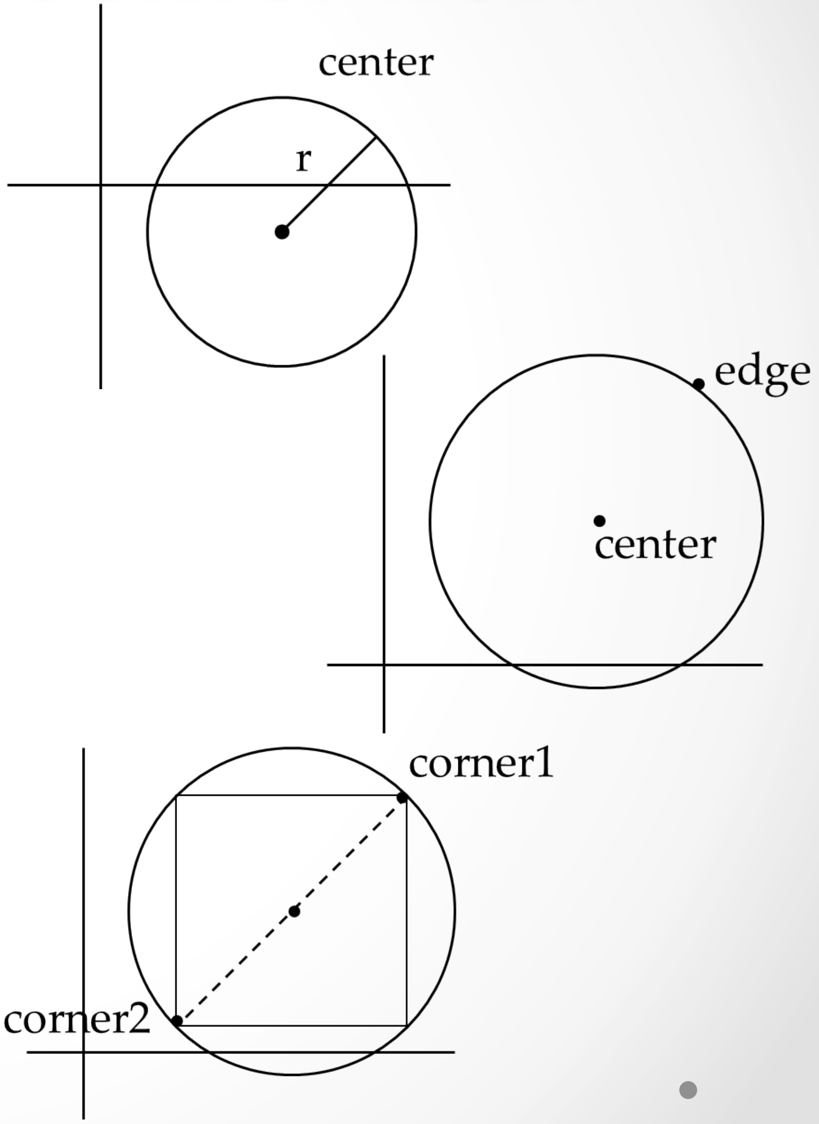
- Center
- Radius

What are some derived fields?

- Circumference
- Area

How can we implement this?

- #1: Center, radius
- #2: Center, edge
- #3: Corner of diameter



Circle Implementation 1

```
public class Circle1 {
       private Point center;
       private double rad;

       // Abstraction function:
       // AF(this) = a circle c such that
       //     c.center =
       //     c.radius =

       // Rep invariant:
       //

       //  ...
}
```

Circle Implementation 1

```
public class Circle1 {
       private Point center;
       private double rad;

       // Abstraction function:
       // AF(this) = a circle c such that
       //     c.center = this.center
       //     c.radius = this.rad

       // Rep invariant:
       // center != null && rad > 0

       //  ...
}
```

Circle Implementation 2

```
public class Circle2 {
       private Point center;
       private Point edge;

       // Abstraction function:
       // AF(this) = a circle c such that
       //     c.center =
       //     c.radius =


       // Rep invariant:
       //

       //    ...
}
```

Circle Implementation 2

```
public class Circle2 {
      private Point center;
      private Point edge;

      // Abstraction function:
      // AF(this) = a circle c such that
      //      c.center = this.center
      //     c.radius = sqrt((center.x-edge.x)^2 + (center.y-
      edge.y)^2)

      // Rep invariant:
      // center != null && edge ! null && !center.equals
(edge)

      //     ...
}
```

Circle Implementation 3

```
public class Circle3 {
       private Point corner1, corner2;

       // Abstraction function:
       // AF(this) = a circle c such that
       //     c.center =

       //     c.radius =


       // Rep invariant:
       //


       //    ...
}
```

Circle Implementation 3

```
public class Circle3 {
      private Point corner1, corner2;

      // Abstraction function:
      // AF(this) = a circle c such that
      //     c.center = <(corner1.x + corner2.x) / 2,
      (corner.y + corner2.y) / 2>
      //     c.radius = (1/2)*sqrt((corner1.x-corner2.x)^2 +
      (corner1.y-corner2.y)^2)

      // Rep invariant:
      // corner1 != null && corner2 != null &&      !
corner1.equals(corner2)

      //    ...
}
```

Checking RIs

- Representation invariant should hold before and after every public method
- Write and use `checkRep()`
  - Call before and after public methods
  - OK that it adds extra code
    - Asserts won’t be included on release builds
    - Important for finding bugs

checkRep() Example

```
public class Circle1 {
        private Point center;
        private double rad;

        private void checkRep() throws RuntimeException {
                if (center == null) {
                        throw new RuntimeException(“This does
                                not have a center");
                }

                if (radius <= 0) {
                        throw new RuntimeException(“This
                        triangle has a negative radius”);
                }
        }
}
```

ADT Example: NonNullStringList

```
public class NonNullStringList  {
       // Abstraction function:
       // ??

       // Rep invariant:
       // ??

       public void add(String s) { ... }
       public boolean remove(String s) { ... }
       public String get(int i) { ... }
}
```

NonNullStringList Implementation 1

```
public class NonNullStringList  {
      // Abstraction function:
      // Index i in arr contains the ith element in the list

      // Rep invariant:
      // RI = [0,count-1] != null

      private String[] arr;
      private int count;

      public void add(String s) { ... }
      public boolean remove(String s) { ... }
      public String get(int i) { ... }
}
```

Problems?

NonNullStringList Implementation 2

```
public class NonNullStringList  {
      // Abstraction function:
      // Value in the nth node after head contains the nth
      item in the list

      // Rep invariant:
      // RI = Head has size nodes after it, each whose value
      is non-      null, no cycle in ListNodes

      public int size;
      public ListNode head;

      public void add(String s) { ... }
      public boolean remove(String s) { ... }
      public String get(int i) { ... }
}
```