Abstract Data Types/Interfaces

Pat Morin
COMP2402

Abstract Data Types

- Describes what a data structure *does*:

Abstract Data Types

- Describes what a data structure *does*:
  - supported operations (the *interface*)

Abstract Data Types

- Describes what a data structure *does*:
  - supported operations (the *interface*)
  - meaning of operations (the *semantics*)

Abstract Data Types

- Describes what a data structure *does*:
  - supported operations (the *interface*)
  - meaning of operations (the *semantics*)
- ~~Representation and implementation~~

Abstract Data Types/Interfaces

B. Liskov at the Turing Centenary Celebration (Wikimedia commons)

The List Interface

- Represents a *sequence* of n items

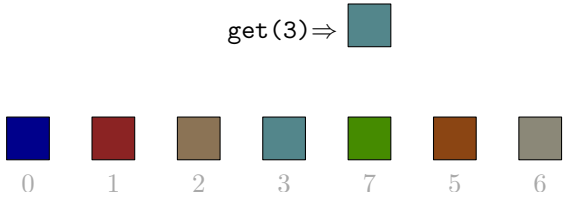
The List Interface

- Represents a *sequence* of n items

- Elements indexed by position, $0, \ldots, n-1$.

The List Interface

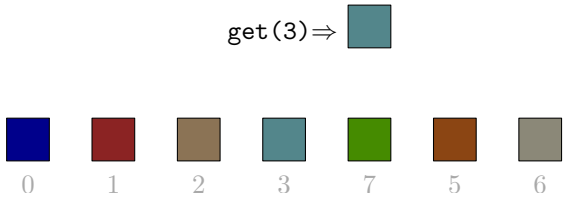
- Represents a *sequence* of n items



- Interface:

The List Interface

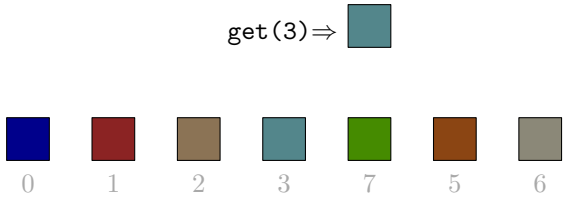
- Represents a *sequence* of n items



- Interface: `size()`

The List Interface

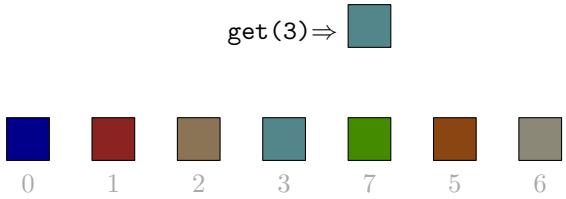
- Represents a *sequence* of n items



- Interface: `size()`, `get(i)`

The List Interface

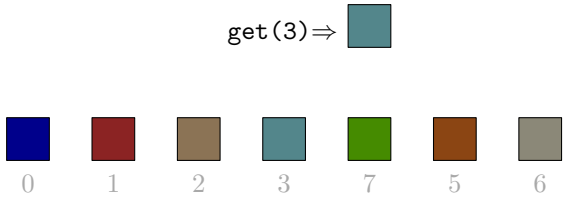
- Represents a *sequence* of n items



- Interface: `size()`, `get(i)`, `set(i,x)`

The List Interface

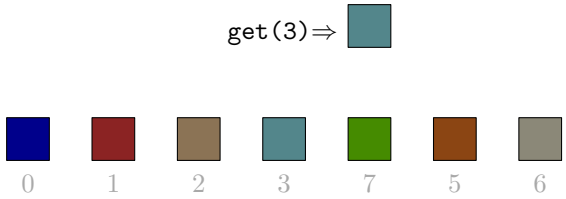
- Represents a *sequence* of n items



- Interface: `size()`, `get(i)`, `set(i,x)`, `add(i,x)`

The List Interface

- Represents a *sequence* of n items



- Interface: `size()`, `get(i)`, `set(i,x)`, `add(i,x)`, `remove(x)`