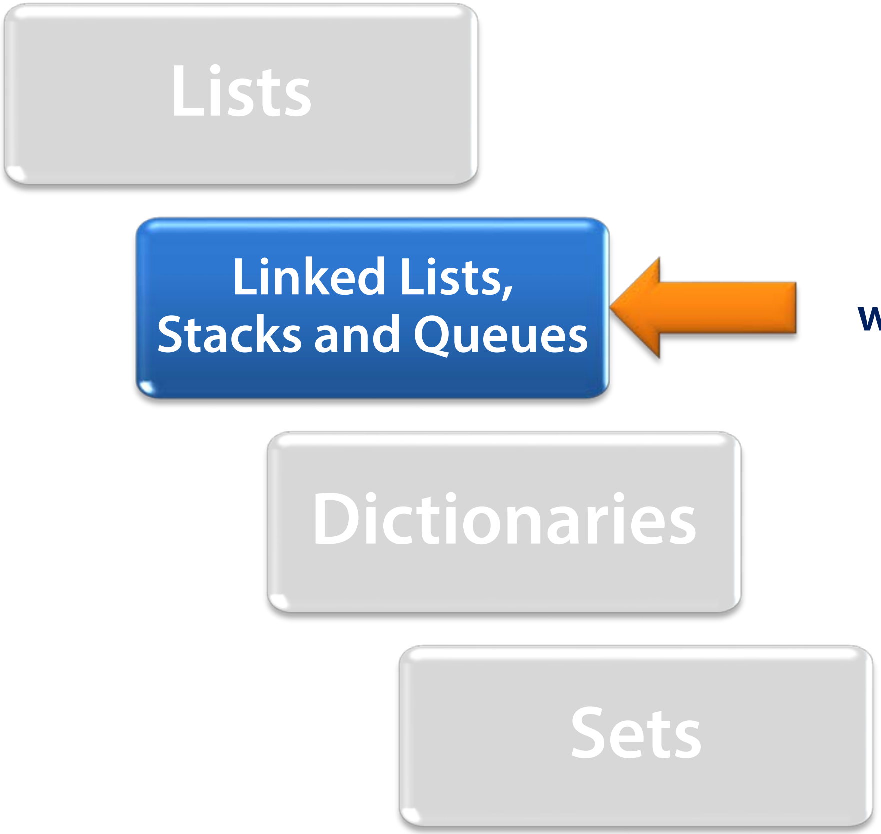
Linked Lists, Stacks and Queues

Simon Robinson
http://TechieSimon.com
@TechieSimon

pluralsight
hardcore developer training



Lists

Linked Lists, Stacks and Queues

Dictionaries

Sets

These are lists but without index-based look-up

Module Overview

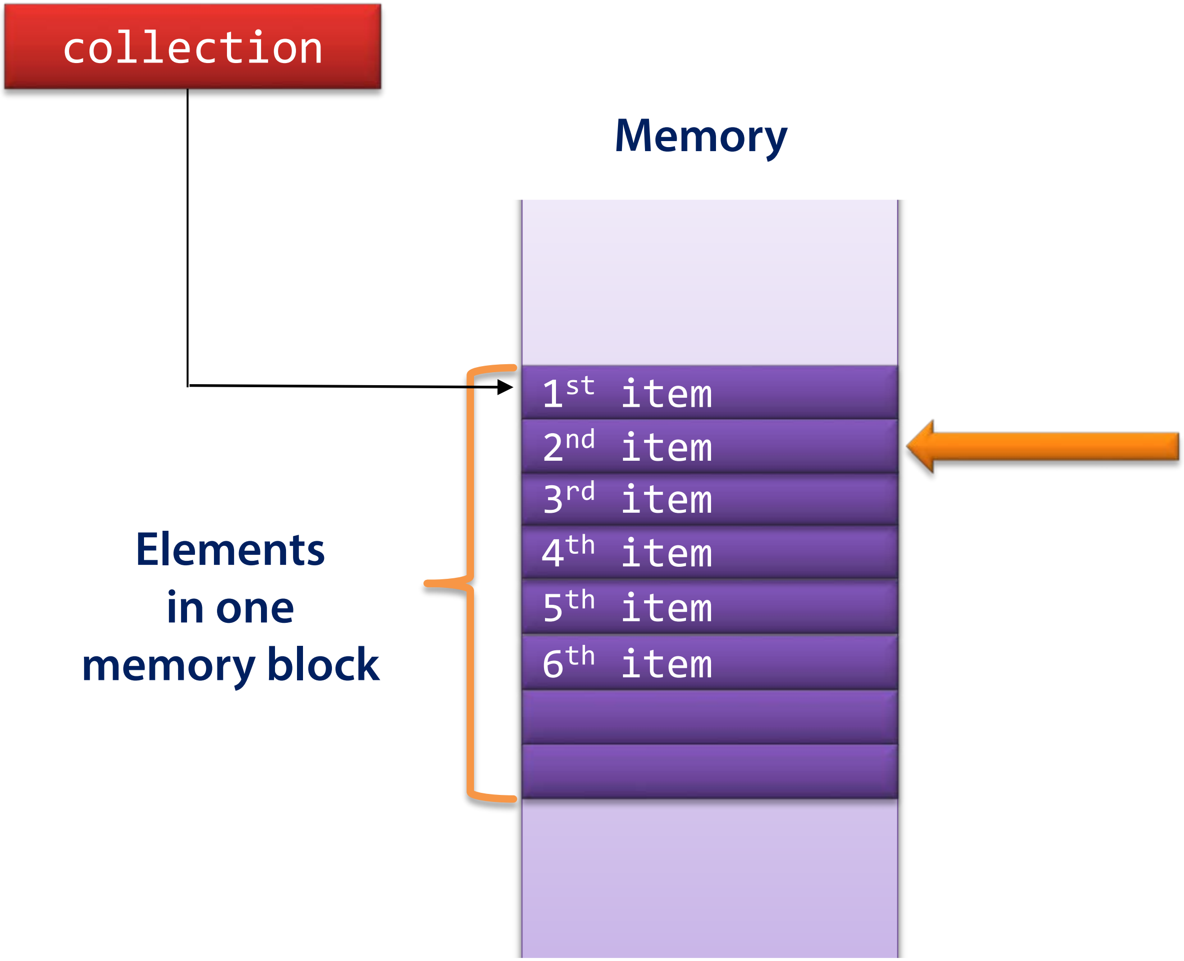
- `LinkedList<T>`
  - List with fast adding/removing elements
- `LinkedListNode<T>`
  - Required to store items in a linked list
- `Stack<T>`
  - First-in last-out list
- `Queue<T>`
  - Remove items in the same order as added (first-in first-out)

```
using System.Collections.Generic;
```

LinkedList<T>

Purpose: Collection that's quick at adding/removing elements

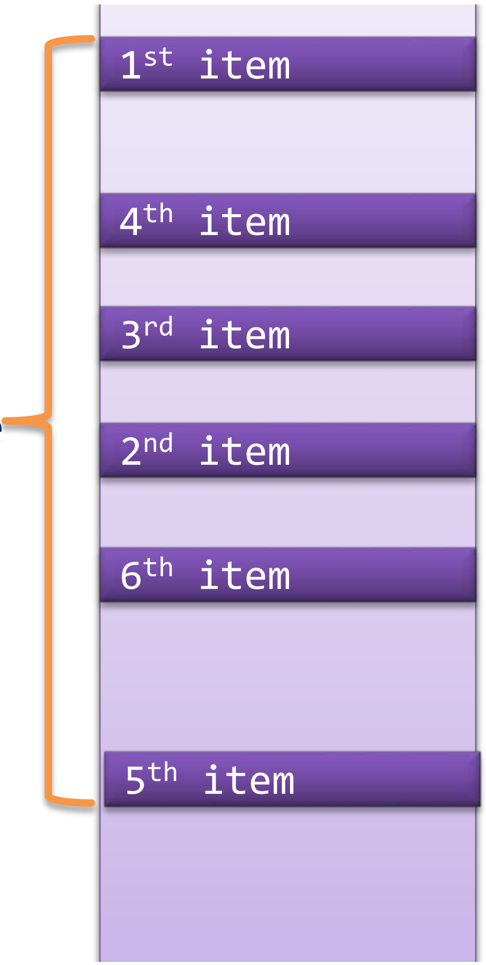
Collections so far:



Elements
in one
memory block

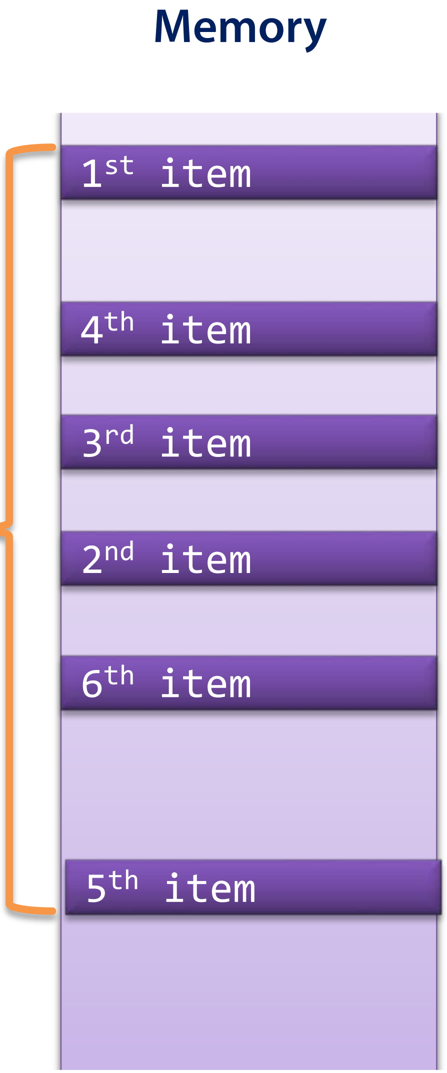
Adding/removing
an item here
means
moving everything
beyond this

Memory



Elements
can be anywhere

Don't need to move
anything!



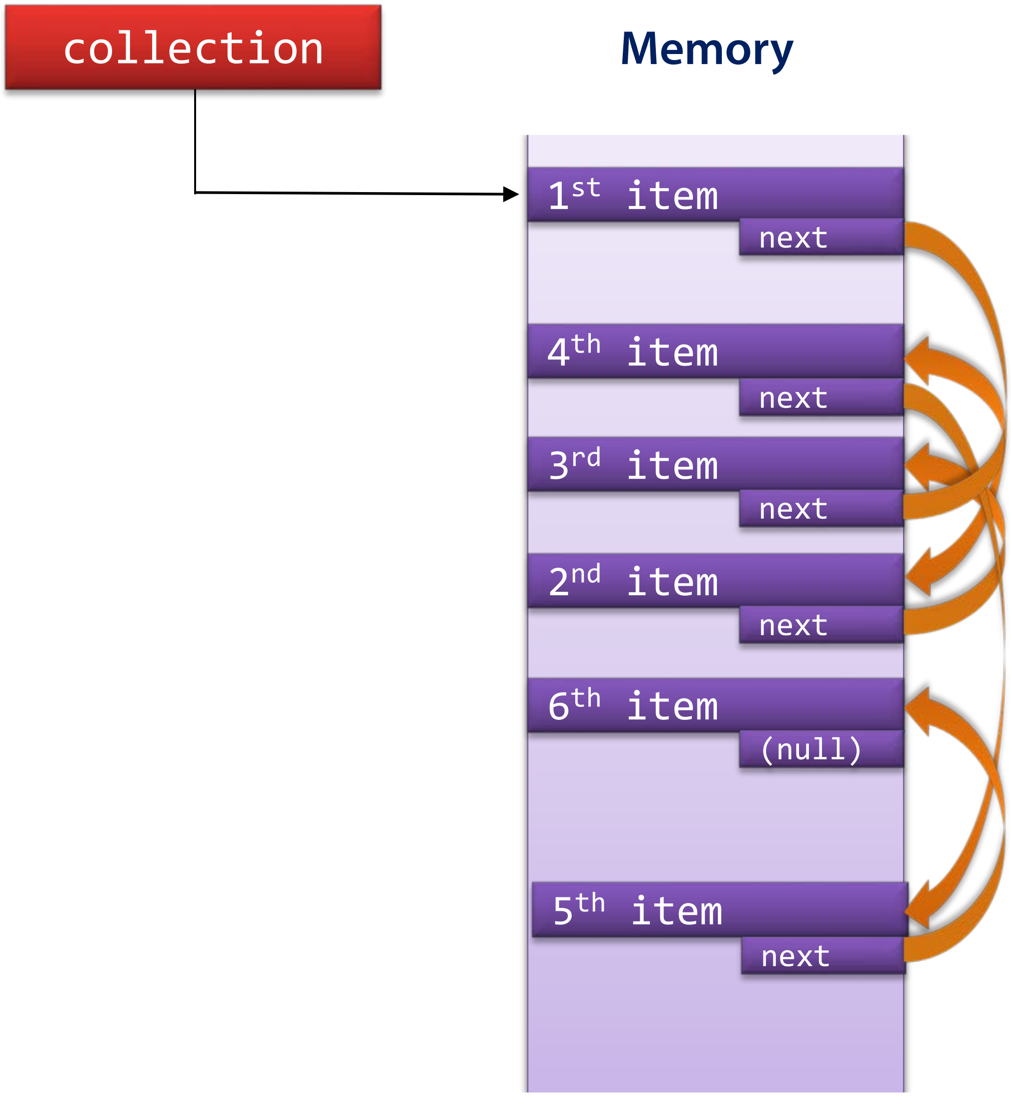
Elements
can be anywhere

Don't need to move
anything!

```
myVar = collection[3];
```

Where is it?

Memory


collection

1st item

next

4th item

next

3rd item

next

2nd item

next

6th item

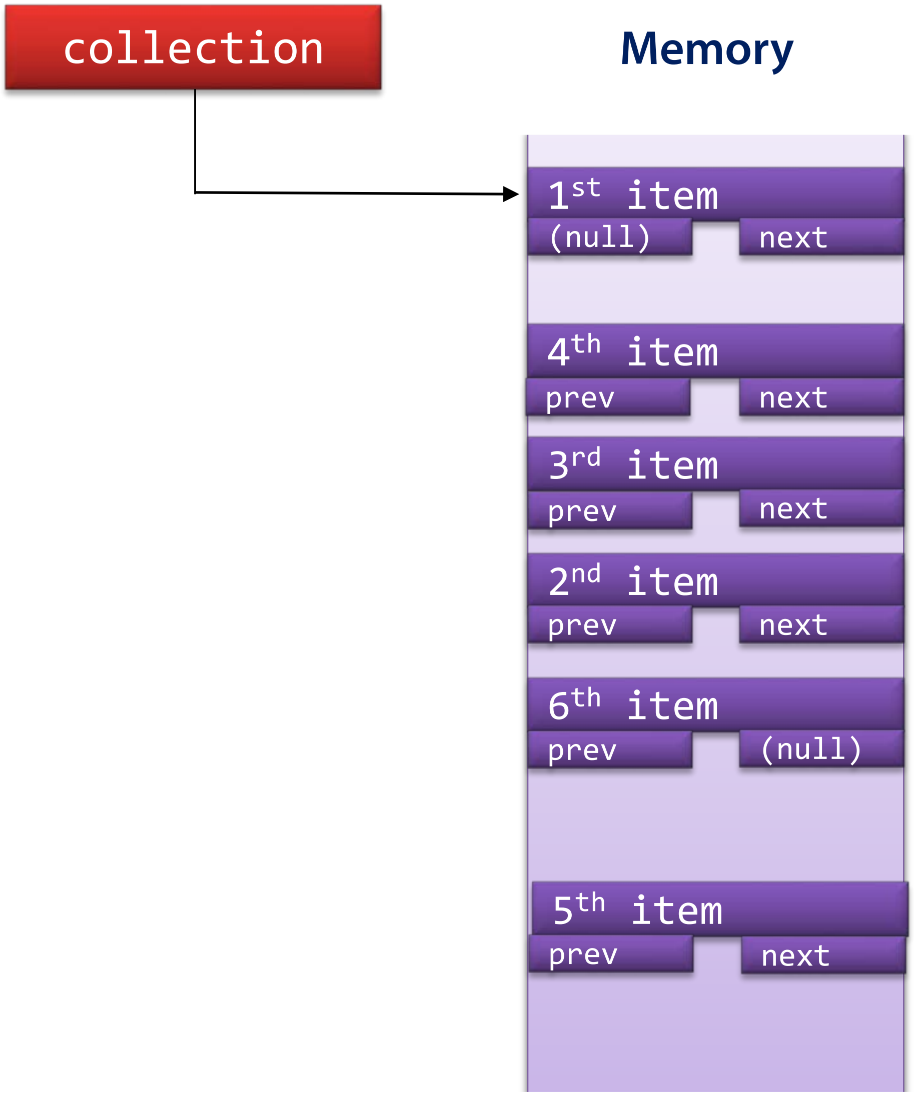
(null)

5th item

next

Can't jump to an element

This is a linked list



collection

Memory

1st item

(null)

next

4th item

prev

next

3rd item

prev

next

2nd item

prev

next

6th item

prev

(null)

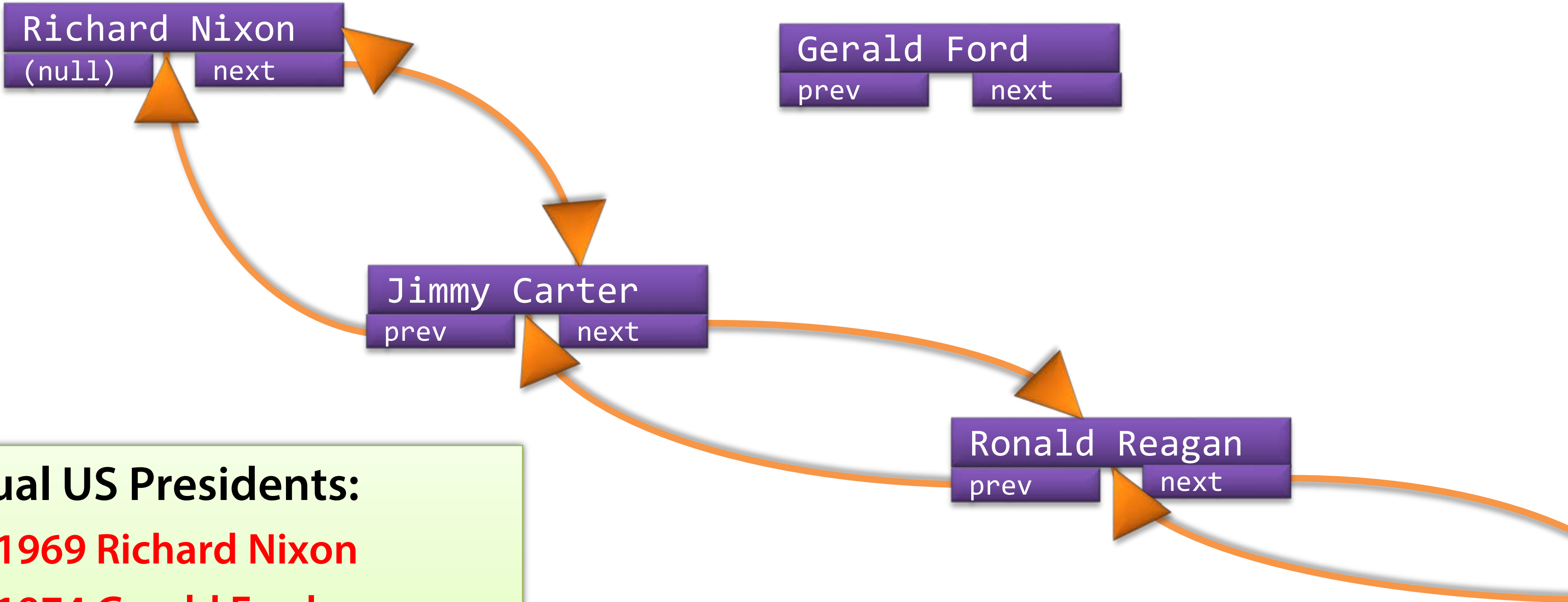
5th item

prev

next

This is a doubly linked list

Insert this:



Actual US Presidents:

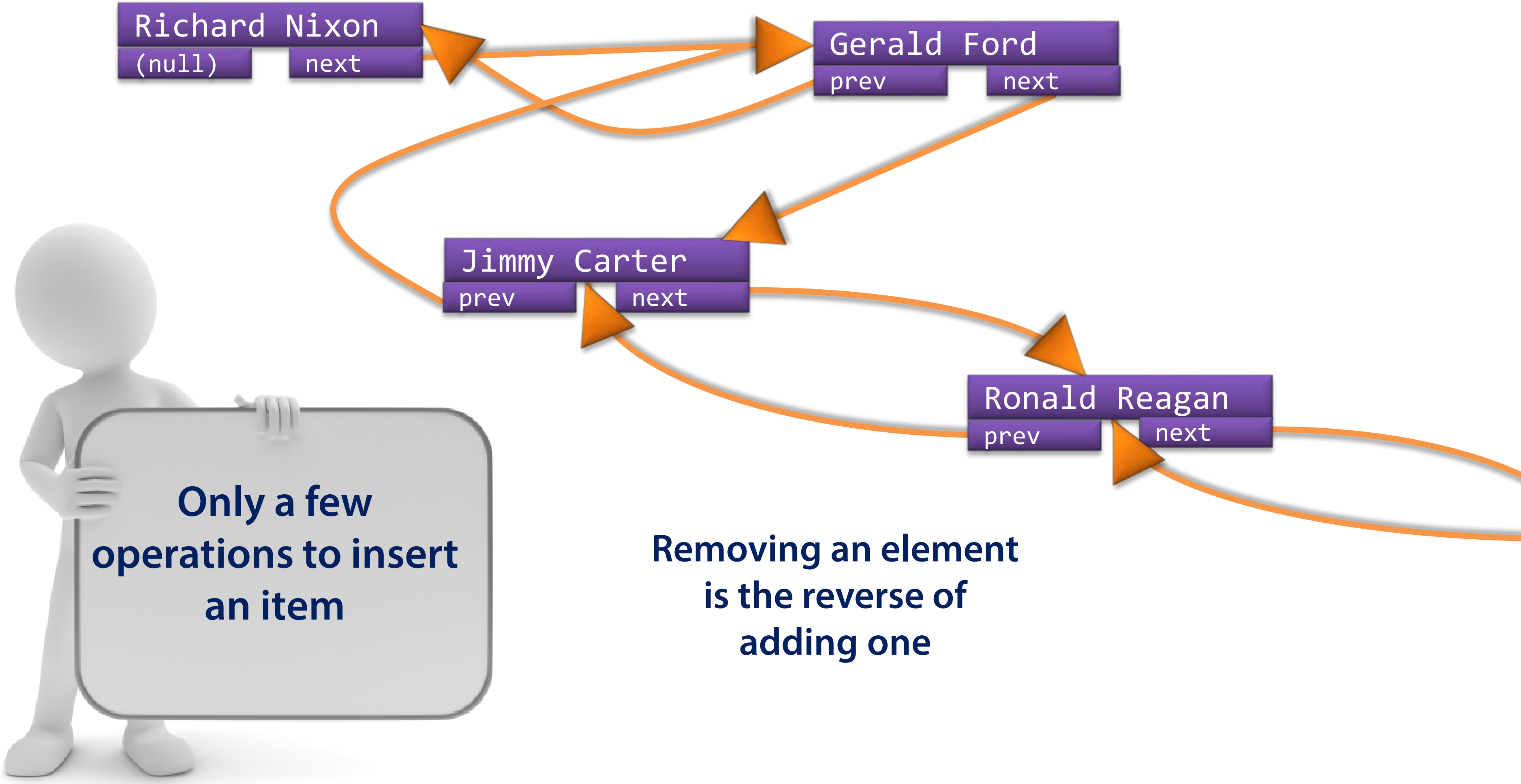
1969 Richard Nixon

1974 Gerald Ford

1977 Jimmy Carter

1981 Ronald Reagan

Insert this:



Only a few operations to insert an item

Removing an element is the reverse of adding one

Linked Lists

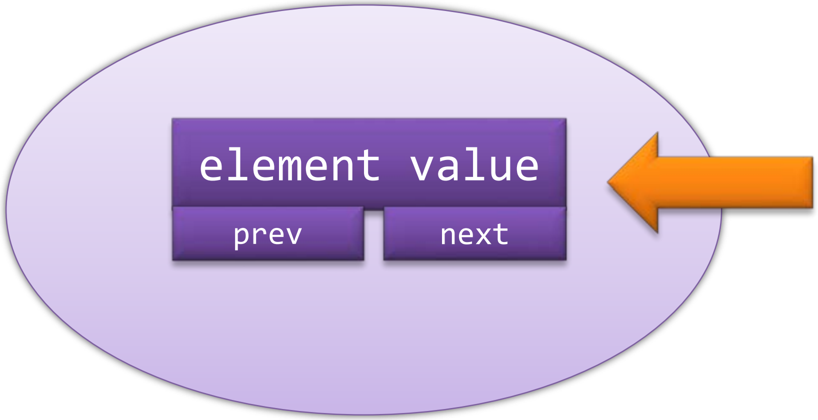
Adding/removing elements is FAST!

Good at enumerating

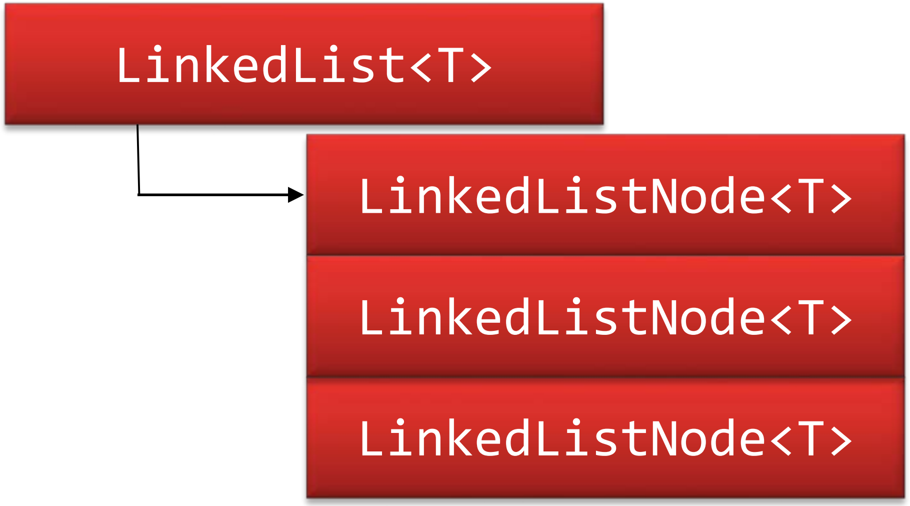
No index-based access to elements

Use lots of memory

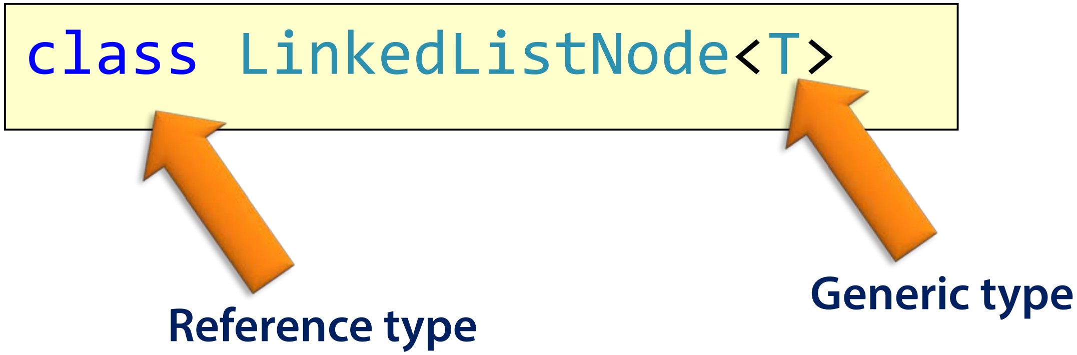
LinkedListNode<T>



Extra info with each element



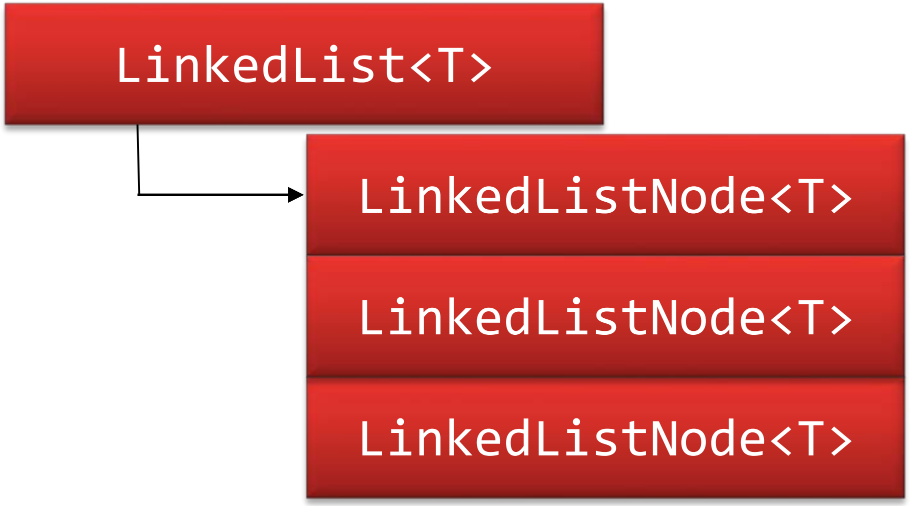
LinkedList<T> is a collection of LinkedListNode<T> not of T



```
class LinkedListNode<T>
```

Reference type

Generic type



LinkedList<T>

LinkedListNode<T>

LinkedListNode<T>

LinkedListNode<T>

`LinkedList<T>` is a collection of `LinkedListNode<T>` not of `T`

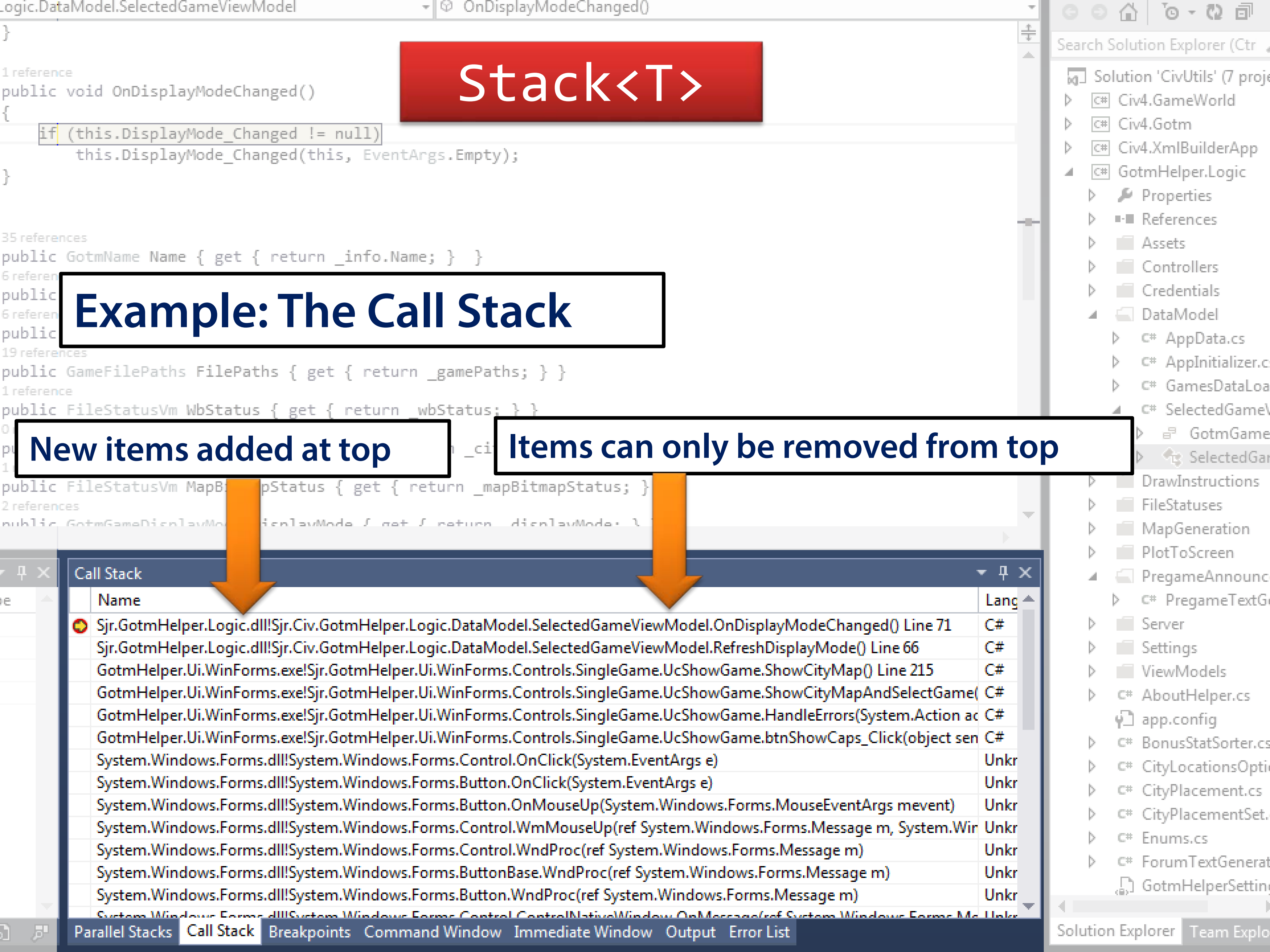
Code Demo

Stack<T>

Example: The Call Stack

New items added at top

Items can only be removed from top



```
}

1 reference
public void OnDisplayModeChanged()
{
    if (this.DisplayMode_Changed != null)
        this.DisplayMode_Changed(this, EventArgs.Empty);
}

35 references
public GotmName Name { get { return _info.Name; }  }
6 referen
public
6 referen
public
19 references
public GameFilePaths FilePaths { get { return _gamePaths; } }
1 reference
public FileStatusVm WbStatus { get { return _wbStatus; } }

public FileStatusVm MapB          pStatus { get { return _mapBitmapStatus; }
2 references
```

Name	Lang
Sjr.GotmHelper.Logic.dll!Sjr.Civ.GotmHelper.Logic.DataModel.SelectedGameViewModel.OnDisplayModeChanged() Line 71	C#
Sjr.GotmHelper.Logic.dll!Sjr.Civ.GotmHelper.Logic.DataModel.SelectedGameViewModel.RefreshDisplayMode() Line 66	C#
GotmHelper.Ui.WinForms.exe!Sjr.GotmHelper.Ui.WinForms.Controls.SingleGame.UcShowGame.ShowCityMap() Line 215	C#
GotmHelper.Ui.WinForms.exe!Sjr.GotmHelper.Ui.WinForms.Controls.SingleGame.UcShowGame.ShowCityMapAndSelectGame(	C#
GotmHelper.Ui.WinForms.exe!Sjr.GotmHelper.Ui.WinForms.Controls.SingleGame.UcShowGame.HandleErrors(System.Action ac	C#
GotmHelper.Ui.WinForms.exe!Sjr.GotmHelper.Ui.WinForms.Controls.SingleGame.UcShowGame.btnShowCaps_Click(object sen	C#
System.Windows.Forms.dll!System.Windows.Forms.Control.OnClick(System.EventArgs e)	Unkr
System.Windows.Forms.dll!System.Windows.Forms.Button.OnClick(System.EventArgs e)	Unkr
System.Windows.Forms.dll!System.Windows.Forms.Button.OnMouseUp(System.Windows.Forms.MouseEventArgs mevent)	Unkr
System.Windows.Forms.dll!System.Windows.Forms.Control.WmMouseUp(ref System.Windows.Forms.Message m, System.Wir	Unkr
System.Windows.Forms.dll!System.Windows.Forms.Control.WndProc(ref System.Windows.Forms.Message m)	Unkr
System.Windows.Forms.dll!System.Windows.Forms.ButtonBase.WndProc(ref System.Windows.Forms.Message m)	Unkr
System.Windows.Forms.dll!System.Windows.Forms.Button.WndProc(ref System.Windows.Forms.Message m)	Unkr
System.Windows.Forms.dll!System.Windows.Forms.Control.ControlNativeWindow.OnMessage(ref System.Windows.Forms.Me	Unkr

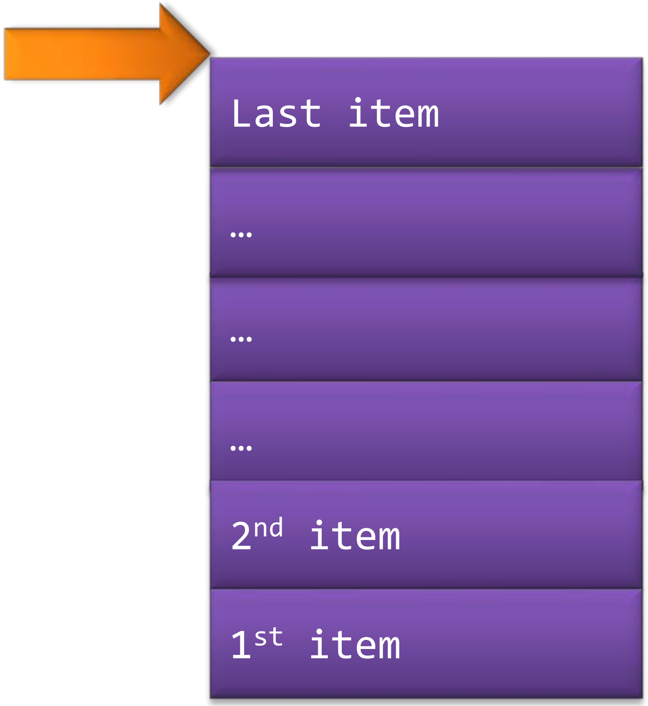
Stack<T>

Last-in First-out collection

Push an item

Pop an item

Items can only be added or removed here



Last item

…

…

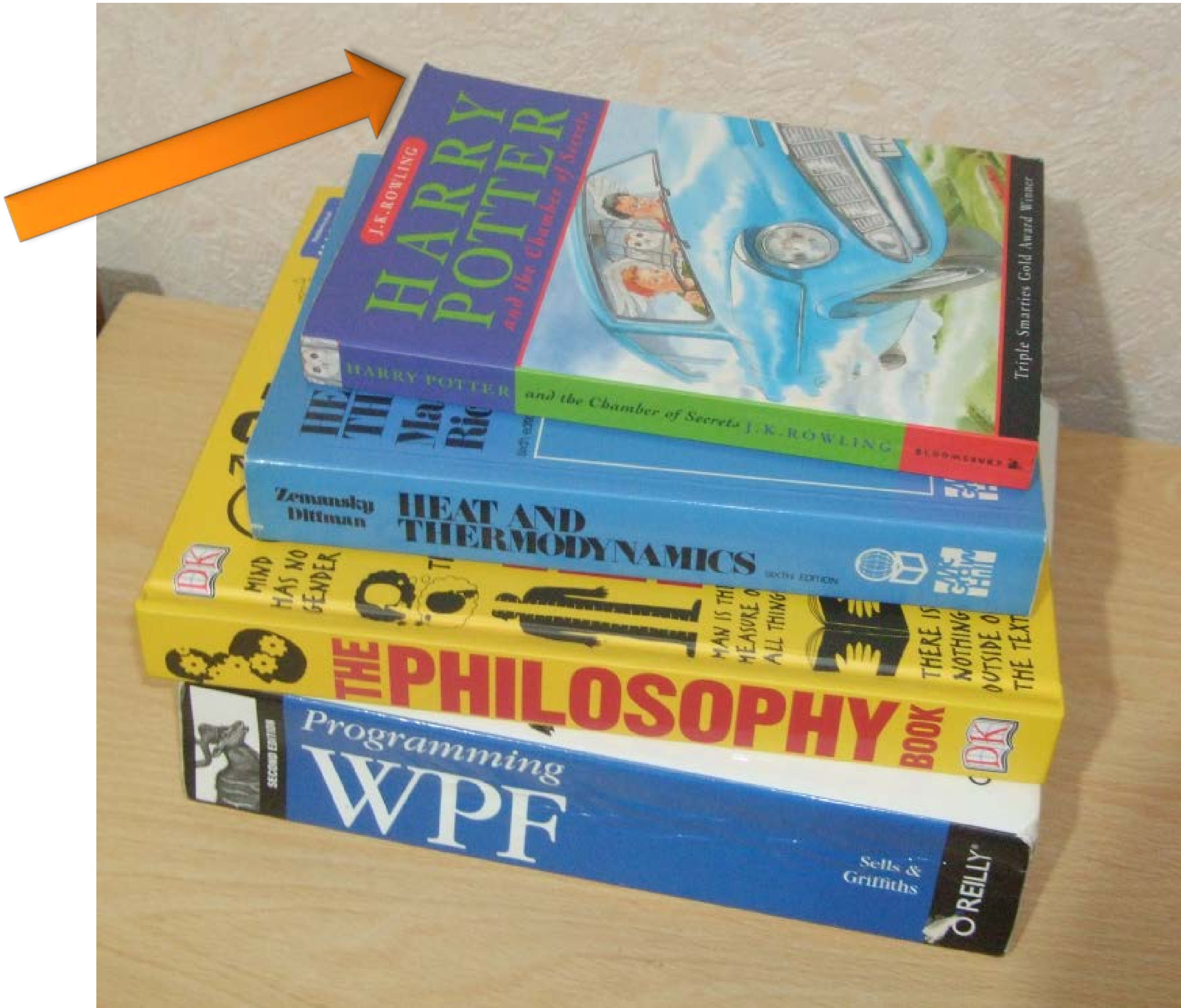
…

2nd item

1st item

Stack<T>

Example: Pile of books



Books can only be added or removed here

Code Demo

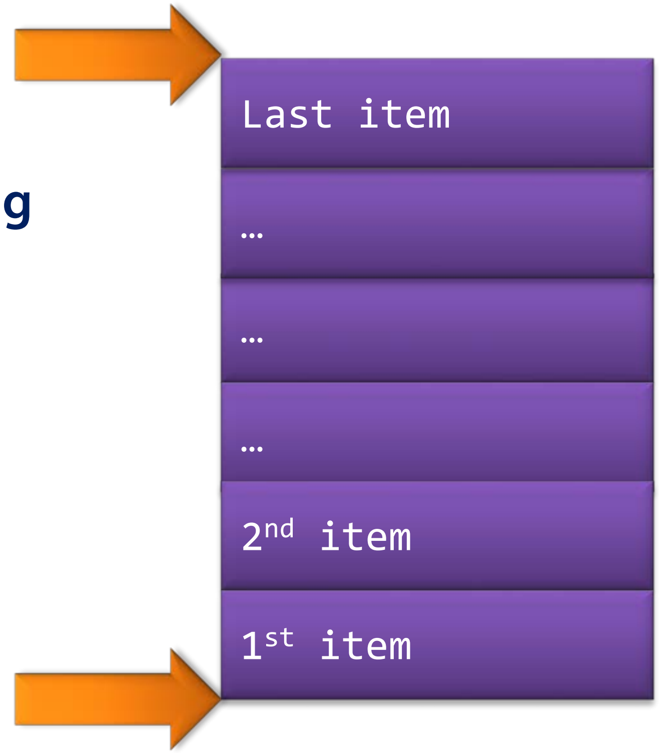
Queue<T>

First-in First-out collection

Items can only be added here

Very useful for storing tasks to be done

Unlike a stack – can only remove items here



Enqueue an item

Dequeue an item

Code Demo

Summary

- `LinkedList<T>`
  - □ `Very efficient adding and removing elements`
- `LinkedListNode<T>`
  - □ Required for adding items to a linked list
- `Stack<T>`
  - □ Last-in first-out list
- `Queue<T>`
  - □ Removes items in order (first-in first-out)