



# Lecture 10: Lists and Sequences

(Sections 10.0-10.2, 10.4-10.6, 10.8-10.13)

## CS 1110

### Introduction to Computing Using Python



[E. Andersen, A. Bracy, D. Gries, L. Lee, S. Marschner, C. Van Loan, W. White]

## Sequences: Lists of Values

String	List																						
<ul style="list-style-type: none"> <li><code>s = 'abc d'</code></li> </ul> <table border="1"> <tr><td>0</td><td>1</td><td>2</td><td>3</td><td>4</td></tr> <tr><td>a</td><td>b</td><td>c</td><td></td><td>d</td></tr> </table> <ul style="list-style-type: none"> <li>Put characters in quotes                             <ul style="list-style-type: none"> <li>Use <code>\</code> for quote character</li> </ul> </li> <li>Access characters with <code>[]</code> <ul style="list-style-type: none"> <li><code>s[0]</code> is 'a'</li> <li><code>s[5]</code> causes an error</li> <li><code>s[0:2]</code> is 'ab' (excludes e)</li> <li><code>s[2:]</code> is 'c d'</li> </ul> </li> </ul>	0	1	2	3	4	a	b	c		d	<ul style="list-style-type: none"> <li><code>x = [5, 6, 5, 9, 15, 23]</code></li> </ul> <table border="1"> <tr><td>0</td><td>1</td><td>2</td><td>3</td><td>4</td><td>5</td></tr> <tr><td>5</td><td>6</td><td>5</td><td>9</td><td>15</td><td>23</td></tr> </table> <ul style="list-style-type: none"> <li>Put values inside <code>[]</code> <ul style="list-style-type: none"> <li>Separate by commas</li> </ul> </li> <li>Access <b>values</b> with <code>[]</code> <ul style="list-style-type: none"> <li><code>x[0]</code> is 5</li> <li><code>x[6]</code> causes an error</li> <li><code>x[0:2]</code> is [5, 6] (excludes 2<sup>nd</sup> 5)</li> <li><code>x[3:]</code> is [9, 15, 23]</li> </ul> </li> </ul>	0	1	2	3	4	5	5	6	5	9	15	23
0	1	2	3	4																			
a	b	c		d																			
0	1	2	3	4	5																		
5	6	5	9	15	23																		

Sequence is a name we give to both

## Lists Have Methods Similar to String

`x = [5, 6, 5, 9, 15, 23]`

- `<list>.index(<value>)`
  - Return position of the value
  - ERROR** if value is not there
  - `x.index(9)` evaluates to 3
- `<list>.count(<value>)`
  - Returns number of times value appears in list
  - `x.count(5)` evaluates to 2

But to get the length of a list you use a function, not a class method:

`len(x)`  
~~`x.len()`~~

## Representing Lists

**Wrong:**

Global Space

~~x = [5, 6, 7, -2]~~

**Correct:**

Global Space

x = id1

Heap Space

id1	list
0	5
1	7
2	4
3	-2

Indices

x = [5, 7, 4, -2]

## Lists vs. Class Objects

List		Objects											
<ul style="list-style-type: none"> <li>Attributes are indexed                             <ul style="list-style-type: none"> <li>Example: <code>x[2]</code></li> </ul> </li> </ul>	<ul style="list-style-type: none"> <li>Attributes are named                             <ul style="list-style-type: none"> <li>Example: <code>p.x</code></li> </ul> </li> </ul>	<p>Global Space</p> <p>p = id3</p>	<p>Heap Space</p> <table border="1"> <tr><td>id3</td><td>Point3</td></tr> <tr><td>x</td><td>1</td></tr> <tr><td>y</td><td>2</td></tr> <tr><td>z</td><td>3</td></tr> </table>	id3	Point3	x	1	y	2	z	3		
id3	Point3												
x	1												
y	2												
z	3												
<p>Global Space</p> <p>x = id2</p>	<p>Heap Space</p> <table border="1"> <tr><td>id2</td><td>list</td></tr> <tr><td>0</td><td>5</td></tr> <tr><td>1</td><td>7</td></tr> <tr><td>2</td><td>4</td></tr> <tr><td>3</td><td>-2</td></tr> </table>	id2	list	0	5	1	7	2	4	3	-2		
id2	list												
0	5												
1	7												
2	4												
3	-2												

## List Assignment

- Format:**

```
<var>[<index>] = <value>
```

  - Reassign at index
  - Affects folder contents
  - Variable is unchanged

```
x = [5, 7, 4, -2]
x[1] = 8
s = "Hello!"
s[0] = 'J'
```

TypeError: 'str' object does not support item assignment

Global Space

x = id1

s = "Hello!"

Heap Space

id1	list
0	5
1	8
2	4
3	-2

- Strings cannot do this
  - Strings are **immutable**

### 1st Clicker Question

- Execute the following:
 

```
>>> x = [5, 6, 5, 9, 10]
>>> x[3] = -1
>>> x.insert(1, 2)
```
- What is x[4]?

A: 10  
 B: 9  
 C: -1  
 D: **ERROR**  
 E: I don't know

16

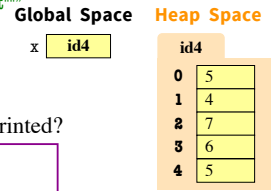
### 2nd Clicker Question

```
def swap(b, h, k):
    """Procedure swaps b[h] and b[k] in b
    Precondition: b is a mutable list, h
    and k are valid positions in the list"""
    1 temp= b[h]
    2 b[h]= b[k]
    3 b[k]= temp
```

x = [5,4,7,6,5]  
 swap(x, 3, 4)  
 print x[3]

What gets printed?

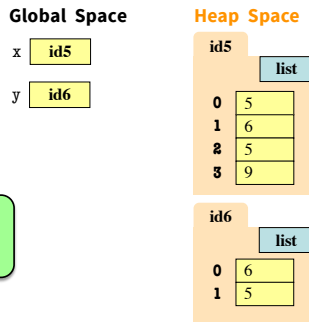
A: 5  
 B: 6  
 C: Something else  
 D: I don't know



20

### List Slices Make Copies

x = [5, 6, 5, 9]  
 y = x[1:3]



copy means  
 new folder

26

### 3rd Clicker Question

- Execute the following:
 

```
>>> x = [5, 6, 5, 9, 10]
>>> y = x[1:]
>>> y[0] = 7
```
- What is x[1]?

A: 7  
 B: 5  
 C: 6  
 D: **ERROR**  
 E: I don't know

27

### 4th Clicker Question

- Execute the following:
 

```
>>> x = [5, 6, 5, 9, 10]
>>> y = x
>>> y[1] = 7
```
- What is x[1]?

A: 7  
 B: 5  
 C: 6  
 D: **ERROR**  
 E: I don't know

29

### Lists and Expressions / 5th Clicker Q

- List brackets [] can contain expressions
- This is a list **expression**
  - Python must evaluate it
  - Evaluates each expression
  - Puts the value in the list
- Example:
 

```
>>> a = [1+2, 3+4, 5+6]
>>> a
[3, 7, 11]
```
- Execute the following:
 

```
>>> a = 5
>>> b = 7
>>> x = [a, b, a+b]
```
- What is x[2]?

A: 'a+b'  
 B: 12  
 C: 57  
 D: **ERROR**  
 E: I don't know

31