CSI31 Lecture 4

<u>Topics</u>:

2.6 Definite Loops
2.7 Future Value Example

2.6 Definite Loops

recall chaotic function (from lecture 2):

```
for i in range(10):
    x = 3.9 * x * (1-x)
    print(x)
```

loop body - body of the loop

- we use loops to execute a sequence of statements several times in succession.

iteration is one execution of that sequence of statements

definite loop - is a simplest kind of loop. At the point in the program when the loop begins, Python knows how many times to *iterate* the body of the loop.

The example given above is the example of the definite loop, and is called *counted loop*.

2.6 Definite Loops

Example:

```
x = 2
for i in range(4):
      x = x+i*i
      print(x)
```

2.6 Definite Loops

Example:

[0,1,2,3]

```
x = 2
for i in range(4):
      x = x+i*i
      print(x)
```

2.6 Definite Loops

Example:

x 2

```
x = 2
for i in range(4):
    x = x+i*i
    print(x)
```

2.6 Definite Loops

Example:

[0,1,2,3]

```
x = 2
for i in range(4):
    x = x+i*i
    print(x)
```

x $\boxed{2}$

2.6 Definite Loops

Example:

[0,1,2,3]

```
x = 2
for i in range(4):
  → x = x+i*i
    print(x)
```

1st iteration

x	2
i	0

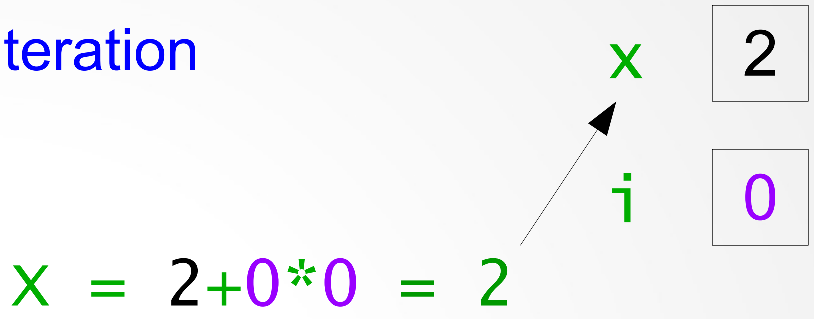
2.6 Definite Loops

Example:

[0,1,2,3]

```
x = 2
for i in range(4):
    x = x+i*i
    print(x)
```

1st iteration

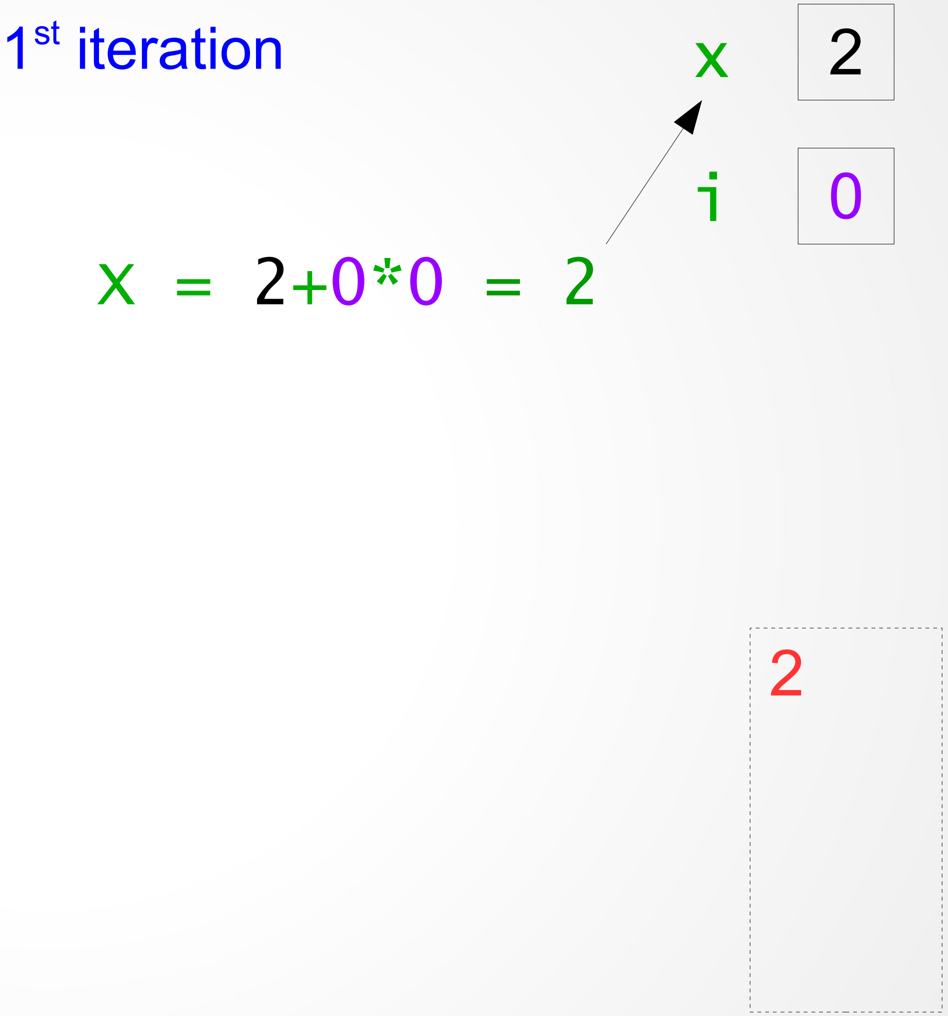


2.6 Definite Loops

Example:

[0,1,2,3]

```
x = 2
for i in range(4):
    x = x+i*i
    print(x)
```



1st iteration

x = 2+0*0 = 2

2.6 Definite Loops

Example:

[0,1,2,3]

```
x = 2
for i in range(4):
    x = x+i*i
    print(x)
```

2nd iteration

x 2

i 1

2

2.6 Definite Loops

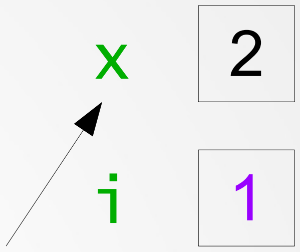
Example:

[0,1,2,3]

```
x = 2
for i in range(4):
    x = x+i*i
    print(x)
```

2nd iteration

x = 2+1*1 = 3



2

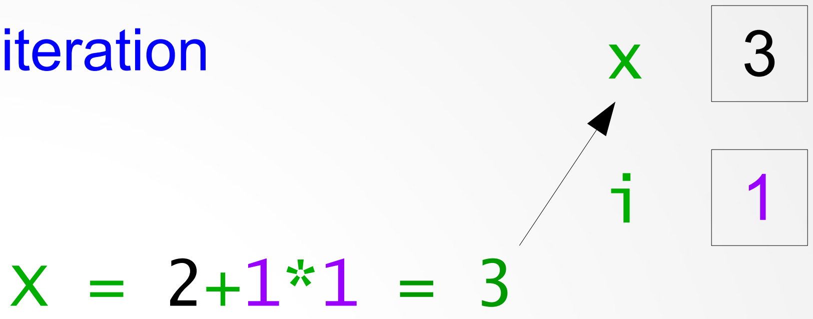
2.6 Definite Loops

Example:

[0,1,2,3]

```
x = 2
for i in range(4):
    x = x+i*i
  → print(x)
```

2nd iteration



$x = 2+1*1 = 3$

```
2
3
```

2.6 Definite Loops

Example:

[0,1,2,3]

```
  x = 2
→ for i in range(4):
      x = x+i*i
      print(x)
```

3rd iteration

x 3

i 2

2
3

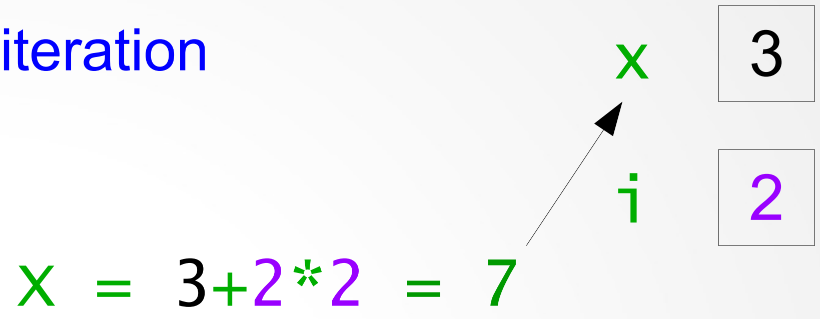
2.6 Definite Loops

Example:

[0,1,2,3]

```
x = 2
for i in range(4):
    x = x+i*i
    print(x)
```

3rd iteration



x = 3+2*2 = 7

2
3

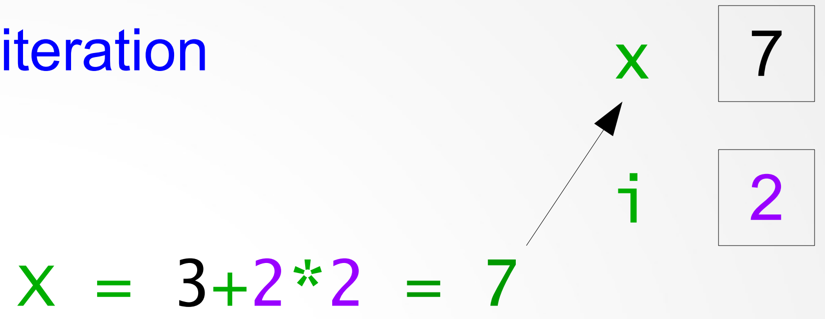
2.6 Definite Loops

Example:

[0,1,2,3]

```
x = 2
for i in range(4):
    x = x+i*i
    print(x)
```

3rd iteration


$x = 3+2*2 = 7$

x 7

i 2

2
3
7

2.6 Definite Loops

Example:

[0,1,2,3]

```
x = 2
for i in range(4):
    x = x+i*i
    print(x)
```

4th iteration

x 7

i 3

```
2
3
7
```

2.6 Definite Loops

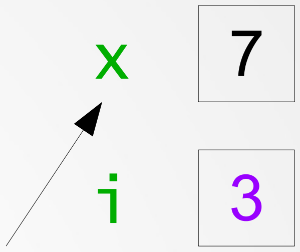
Example:

[0,1,2,3]

```
x = 2
for i in range(4):
    x = x+i*i
    print(x)
```

4th iteration

x = 7+3*3 = 16



```
2
3
7
```

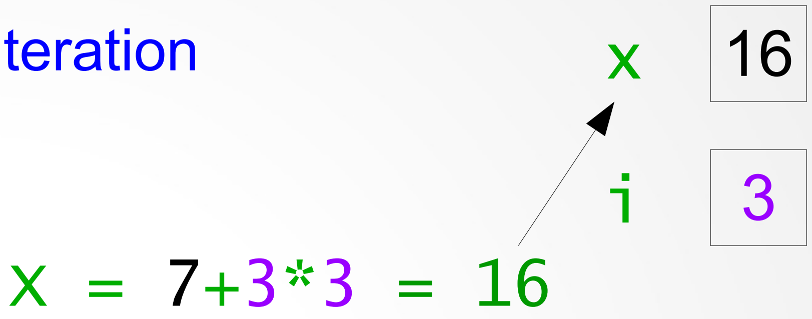
2.6 Definite Loops

Example:

[0,1,2,3]

```
x = 2
for i in range(4):
    x = x+i*i
    print(x)
```

4th iteration



x = 7+3*3 = 16

x 16

i 3

```
2
3
7
16
```

2.6 Definite Loops

Example:

4th iteration

```
            [0,1,2,3]
   x = 2
→  for i in range(4):
       x = x+i*i
       print(x)
```

x 16

i 3

```
2
3
7
16
```

2.6 Definite Loops

Example:

[0,1,2,3]

```
x = 2
for i in range(4):
    x = x+i*i
    print(x)
```

4th iteration

x 16

i 3

```
2
3
7
16
```

2.6 Definite Loops

A Python for loop has this general form:

```
    for <var> in <sequence>:
        <body>
```

loop heading

- <body>   is a sequence of Python statements.
- <var> is the *loop index. I*t takes on each successive value in the sequence, and the statements in the body are exectuted once for each value.)
- <sequence> portion often consists of a *list* of values.

Example:

the length of the list determines the number of times the loop will execute

```
y = 1
for counter in [1,2,3,4]:
    y = y + counter
    print(''counter = '', counter, '', y = '', y)
```

2.6 Definite Loops

example:

```
y = 1
for counter in [1,2,3,4]:
    y = y + counter
    print(''counter = '', counter, '', y = '', y)
```

```
counter = 1, y = 2
counter = 2, y = 4
counter = 3, y = 7
counter = 4, y = 11
```

2.6 Definite Loops

Compare two *counted loops*:

```
for i in range(10):
      x = 3.9 * x * (1-x)
      print(x)
```

```
for counter in [1,2,3,4]:
  y = y + counter
  print(''counter = '', counter, '', y = '', y)
```

2.6 Definite Loops

Compare two *counted loops*:

```
for i in range(10):
      x = 3.9 * x * (1-x)
      print(x)
```

```
for counter in [1,2,3,4]:
   y = y + counter
   print(''counter = '', counter, '', y = '', y)
```

`range(10)` is a sequence of 10 numbers (from 0 till 9)

Try to input in the *interactive window*:

```
>>> range(10)
```

range(0,10) - you will get a sequence of values from 0 to 9

2.6 Definite Loops

The `range` function is a built-in Python function (command)

General form of the range function:

```
    range(<expr>)
```

- will produce a sequence of numbers starting from 0 and going up to, but not including, the value of `<expr>`

If you begin to type in `range(` in the interactive window - you'll see a hint:

```
    range([start,] stop[, step]) -> list of integers
```

Try to input the following statements in the *Python shell*:

```
>>> list(range(4,13))
```

```
>>> list(range(4, 16, 2))
```

and see the result.

2.6 Definite Loops

Statements like *for loops* are called *control structures* because they control the execution of other parts of the program.

Some programmers find it useful to think of control structures in terms of pictures called *flowcharts*.

2.6 Definite Loops

Statements like *for loops* are called *control structures* because they control the execution of other parts of the program.

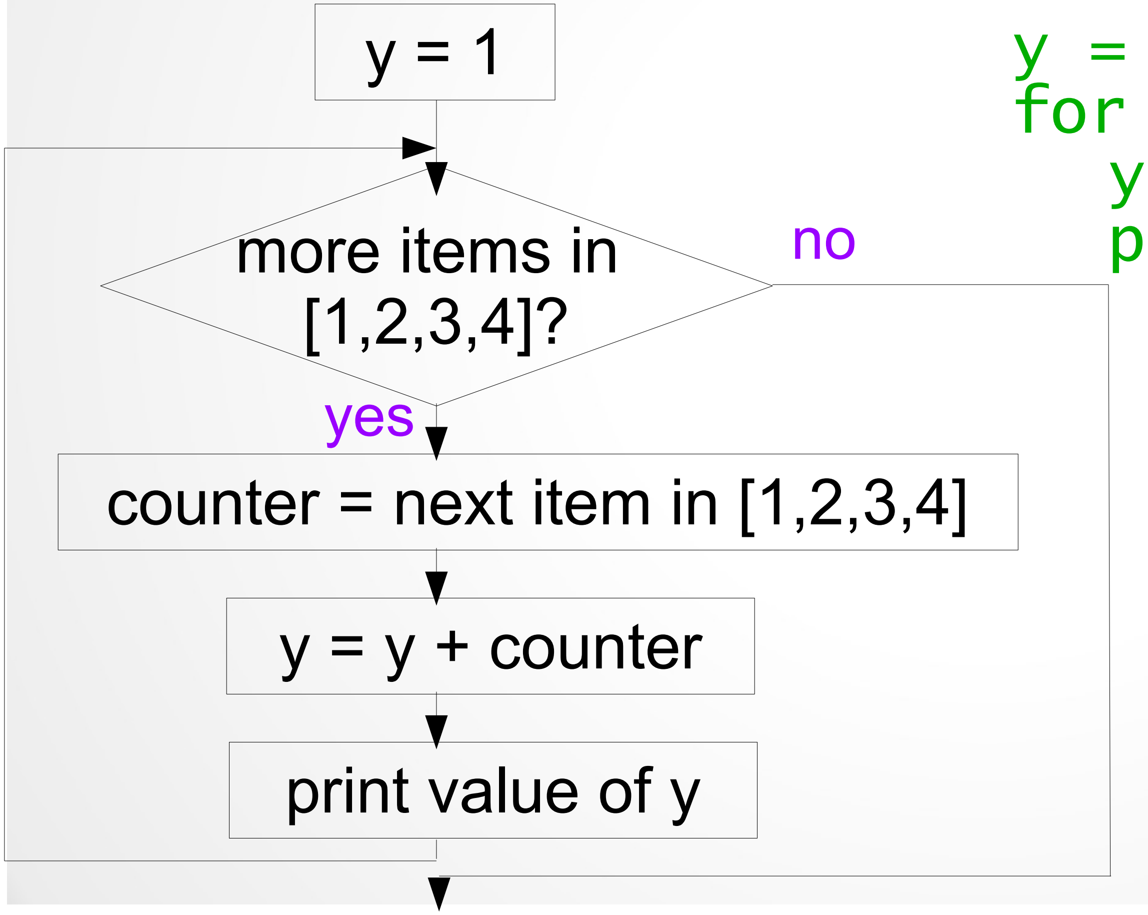
Some programmers find it useful to think of control structures in terms of pictures called *flowcharts*.

```
y = 1
for counter in [1,2,3,4]:
    y = y + counter
    print(y)
```

2.6 Definite Loops

Statements like *for loops* are called *control structures* because they control the execution of other parts of the program.

Some programmers find it useful to think of control structures in terms of pictures called *flowcharts*.



```
y = 1
for counter in [1,2,3,4]:
   y = y + counter
   print(y)
```

2.7 Example program: future value

Let’s develop a program to determine the future value of an investment.

Money is deposited in a bank account (the initial principal).

It earns interest (APY – annual percentage yield).
This is the percent of the principal that will be added to the principal in one year.

Example: For $100 and 3% APY, in a year we will get $103 = $100 + 3% of $100 = $100 + $3.

2.7 Example program: future value

Let’s develop a program to determine the future value of an investment.

Money are deposited in a bank account (the principal).

It earns interest (APY – annual percentage yield).
This is the percent of the principal that will be added to the principal in one year.

Example: For $100 and 3% APY, in a year we will get $103 = $100 + 3% of $100 = $100 + $3.

In another year we will get $103 + 3% of $103 = $103 + $3.09 = $106.09.

2.7 Example program: future value

Let’s develop a program to determine the future value of an investment.

Money are deposited in a bank account (the principal).

It earns interest (APY – annual percentage yield).
This is the percent of the principal that will be added to the principal in one year.

Example: For $100 and 3% APY, in a year we will get $103 = $100 + 3% of $100 = $100 + $3.

In another year we will get $103 + 3% of $103 = $103 + $3.09 = $106.09.

In another year we will get $106.09 + 3% of $106.09 = $103 + $3.1827 = $109.2727.

2.7 Example program: future value

Let’s develop a program to determine the future value of an investment.

Money are deposited in a bank account (the principal).

It earns interest (APY – annual percentage yield).
This is the percent of the principal that will be added to the principal in one year.

Example: For \$100 and 3% APY, in a year we will get \$103 = \$100 + 3% of \$100 = \$100 + \$3.

In another year we will get \$103 + 3% of \$103 = \$103 + \$3.09 = \$106.09.

In another year we will get \$106.09 + 3% of \$106.09 = \$103 + \$3.1827 = \$109.2727.

Etc.

2.7 Example program: future value

Summary: Given the principal and the interest rate (APY), we should be able to calculate the value of the investment any number of years into the future!

2.7 Example program: future value

Summary: Given the principal, the interest rate and the number of compounding periods, we should be able to calculate the value of the investment ten years into the future!

Program name: Future Value

Inputs:

- the amount of money being invested (in dollars)
- the interest rate (APR – annual percentage rate) (in %)
- The number of compounding periods

Output: The value of investment in 10 years.

2.7 Example program: future value

Summary: Given the principal and the interest rate (APY), we should be able to calculate the value of the investment any number of years into the future!

Program name: Future Value

Inputs:

- The initial amount of money being invested (in dollars)
- The interest rate (APY – annual percentage yield) (in %)
- The number of years

Output: The value of investment after the inputted years.

Relationship: value after one year equals:

New principal = old principal + old principal*interest rate

This formula needs to be applied the number of years inputted.