Python I
Getting started with Programming

Jo Grundy

PSDI & ECS Southampton

PSDI
PHYSICAL SCIENCES
DATA INFRASTRUCTURE

Python I - Introduction

Who are we?

PSDI
PHYSICAL SCIENCES
DATA INFRASTRUCTURE

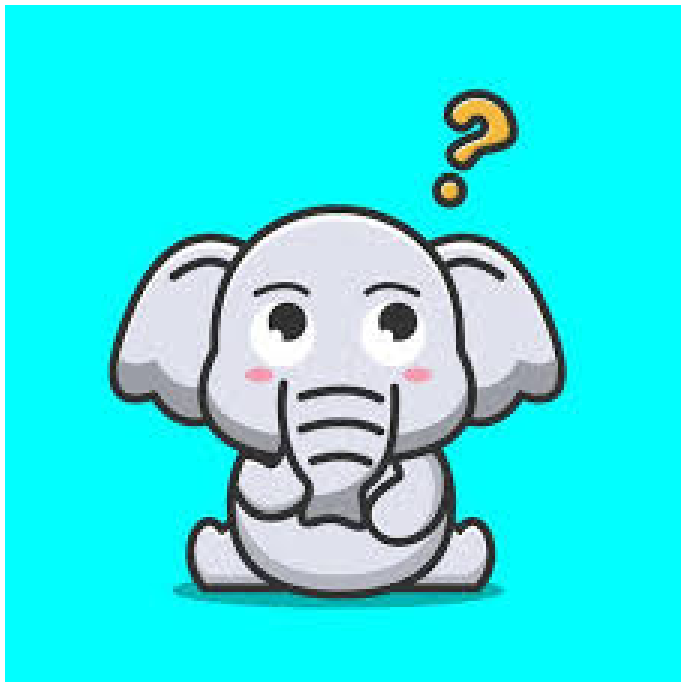
- Dr Jo Grundy (j.grundy@soton.ac.uk) B01 1022B
- PSDI `https://www.psdi.ac.uk/`

PSDI
PHYSICAL SCIENCES
DATA INFRASTRUCTURE

Python I - Introduction

Who are you?

Vevox Quiz 163-489-314



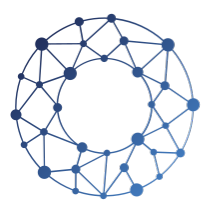
PSDI
PHYSICAL SCIENCES
DATA INFRASTRUCTURE

Python (I) - Functions

Functions allow us to package a set of commands in to a single command

They are defined using the `def` key word

```
def square(number):
    """
    This function takes a number n and
    returns the square of that number
    """
     return number*number
```



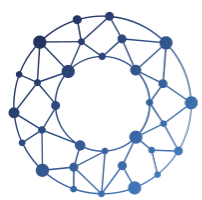
PSDI
PHYSICAL SCIENCES
DATA INFRASTRUCTURE

Python (I) - Functions

Functions allow us to package a set of commands in to a single command

They are defined using the `def` key word

```
def square(number):
    """
    This function takes a number n and
    returns the square of that number
    """
     return number*number
```



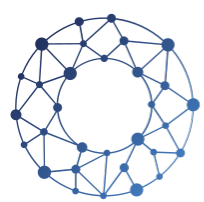
PSDI
PHYSICAL SCIENCES
DATA INFRASTRUCTURE

Python (I) - Functions

Functions allow us to package a set of commands in to a single command

They are defined using the `def` key word

```
def square(number):
    """
    This function takes a number n and
    returns the square of that number
    """
     return number*number
```



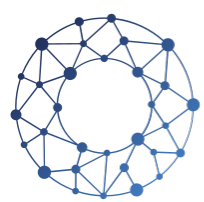
PSDI
PHYSICAL SCIENCES
DATA INFRASTRUCTURE

Python (I) - Functions

Functions allow us to package a set of commands in to a single command

They are defined using the `def` key word

```
def square(number):
    """
    This function takes a number n and
    returns the square of that number
    """
     return number*number
```



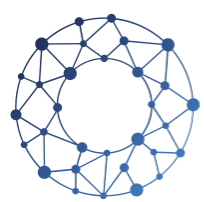
PSDI
PHYSICAL SCIENCES
DATA INFRASTRUCTURE

Python (I) - Functions

Functions allow us to package a set of commands in to a single command

They are defined using the `def` key word

```
def square(number):
    """
    This function takes a number n and
    returns the square of that number
    """
     return number*number
```



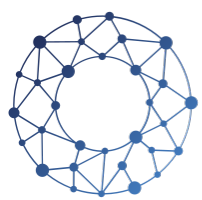
PSDI
PHYSICAL SCIENCES
DATA INFRASTRUCTURE

Python (I) - Functions

Functions allow us to package a set of commands in to a single command

They are defined using the `def` key word

```
def square(number):
    """
    This function takes a number n and
    returns the square of that number
    """
     return number*number
```



PSDI
PHYSICAL SCIENCES
DATA INFRASTRUCTURE

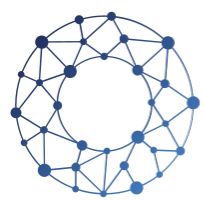
Python (I) - Functions

Functions allow us to package a set of commands in to a single command

They are defined using the `def` key word

```
def square(number):
    """
    This function takes a number n and
    returns the square of that number
    """
     return number*number
```

This is abstraction



PSDI
PHYSICAL SCIENCES
DATA INFRASTRUCTURE

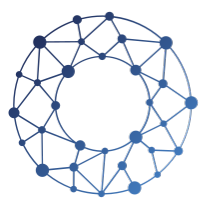
Python (I) - Functions

Functions allow us to package a set of commands in to a single command

They are defined using the `def` key word

```
def square(number):
    """
    This function takes a number n and
    returns the square of that number
    """
     return number*number
```

This is abstraction



PSDI
PHYSICAL SCIENCES
DATA INFRASTRUCTURE

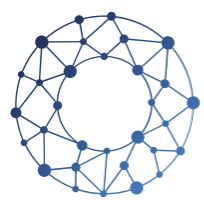
Python (I) - Functions

Functions allow us to package a set of commands in to a single command

They are defined using the `def` key word

```
def square(number):
    """
    This function takes a number n and
    returns the square of that number
    """
     return number*number
```

This is abstraction



PSDI
PHYSICAL SCIENCES
DATA INFRASTRUCTURE

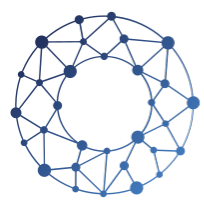
Python (I) - Functions

Functions allow us to package a set of commands in to a single command

They are defined using the `def` key word

```
def square(number):
    """
    This function takes a number n and
    returns the square of that number
    """
     return number*number
```

This is abstraction



PSDI
PHYSICAL SCIENCES
DATA INFRASTRUCTURE

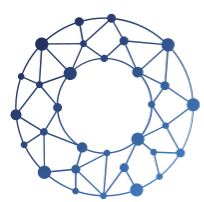
Python (I) - Functions - Questions

PSDI
PHYSICAL SCIENCES
DATA INFRASTRUCTURE

Python (I) - Functions

Scope

- variables such as the arguments and others defined in the function are forgotten afterwards.
- the variable `number` will have no meaning outside the function.
- Try seeing what is in variable `number` after using the function



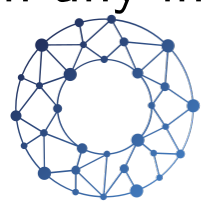
PSDI
PHYSICAL SCIENCES
DATA INFRASTRUCTURE

Python (II) - Programs - Algorithms

Programs use algorithms

- Step 1: Weigh out 200 g flour in to large bowl
- Step 2: Measure out 400 mL milk
- Step 3: Add and egg, and beat it in to the flour with some of the milk
- Step 4: Add 3 more eggs, one by one, beating it in with more of the milk each time
- Step 5: Add all of the milk and beat making sure no lumps
- Step 6: Leave in fridge for 30 mins

This makes pancake batter.
An algorithm is the same, a recipe for what to do with any input.

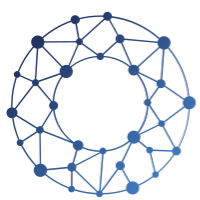


PSDI
PHYSICAL SCIENCES
DATA INFRASTRUCTURE

Python (I) - Functions - Iteration

To see if an item is in a list, you can iterate though it one by one to see if it is there.

```
def find_iter(lst, elem):
    """
    checks to see if item in list
    returns True if it is and False otherwise
    """
    for item in lst:
        if item == elem:
            return True
    return False
```



PSDI
PHYSICAL SCIENCES
DATA INFRASTRUCTURE

Python (I) - Functions - Iteration

To see if an item is in a list, you can iterate though it one by one to see if it is there.

```
def find_iter(lst, elem):
    """
    checks to see if item in list
    returns True if it is and False otherwise
    """
    for item in lst:
        if item == elem:
            return True
    return False
```

How efficient is this?

PSDI
PHYSICAL SCIENCES
DATA INFRASTRUCTURE

Python (I) - Functions - Iteration

To see if an item is in a list, you can iterate though it one by one to see if it is there.

```
def find_iter(lst, elem):
    """
    checks to see if item in list
    returns True if it is and False otherwise
    """
    for item in lst:
        if item == elem:
            return True
    return False
```

How efficient is this? Complexity $\mathcal{O}(n)$ can we do better?

PSDI
PHYSICAL SCIENCES DATA INFRASTRUCTURE

Python (I) - Functions - Iteration

To see if an item is in a list, you can iterate though it one by one to see if it is there.

```
def find_iter(lst, elem):
    """
    checks to see if item in list
    returns True if it is and False otherwise
    """
    for item in lst:
        if item == elem:
            return True
    return False
```

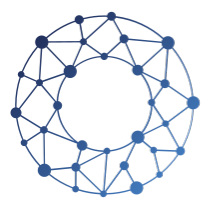
How efficient is this? Complexity $\mathcal{O}(n)$ can we do better? No..

PSDI
PHYSICAL SCIENCES DATA INFRASTRUCTURE

Python (I) - Functions - Iteration

Iteration

What about if the list is sorted? how can we do this better?



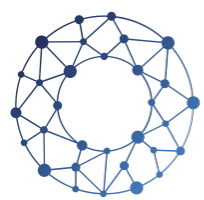
PSDI
PHYSICAL SCIENCES
DATA INFRASTRUCTURE

Python (I) - Functions - Iteration

Iteration

What about if the list is sorted? how can we do this better?

```
def find_iter(lst, elem):
    found = False
    while not found:
        half = len(lst)//2
        if len(lst) ==1:
            return lst[half] == elem
            return True
        if lst[half] > elem:
            lst=lst[half:]
        else lst[half] < elem:
            lst=lst[:half]
```



PSDI
PHYSICAL SCIENCES
DATA INFRASTRUCTURE

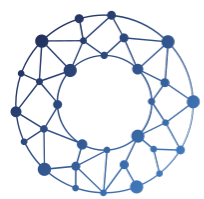
Python (I) - Functions - Iteration

Iteration

What about if the list is sorted? how can we do this better?

```
def find_iter(lst, elem):
    found = False
    while not found:
        half = len(lst)//2
        if len(lst) ==1:
            return lst[half] == elem
            return True
        if lst[half] > elem:
            lst=lst[half:]
        else lst[half] < elem:
            lst=lst[:half]
```

How efficient is this?



PSDI
PHYSICAL SCIENCES
DATA INFRASTRUCTURE

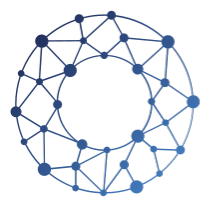
Python (I) - Functions - Iteration

Iteration

What about if the list is sorted? how can we do this better?

```
def find_iter(lst, elem):
    found = False
    while not found:
        half = len(lst)//2
        if len(lst) ==1:
            return lst[half] == elem
            return True
        if lst[half] > elem:
            lst=lst[half:]
        else lst[half] < elem:
            lst=lst[:half]
```

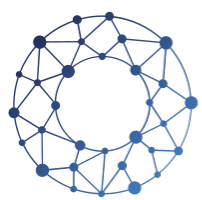
How efficient is this? Complexity $\mathcal{O}(\log n)$



PSDI
PHYSICAL SCIENCES
DATA INFRASTRUCTURE

Python (II) - Programs - Iteration

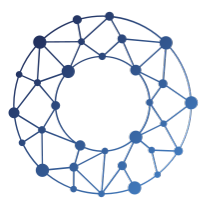
- iteratively multiply a number `n` by `m`



PSDI
PHYSICAL SCIENCES
DATA INFRASTRUCTURE

Python (II) - Programs - Iteration

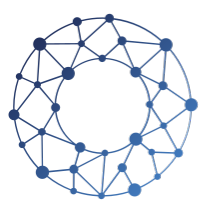
- iteratively multiply a number `n` by `m`
- iteratively find if a string `s` is a palindrome



PSDI
PHYSICAL SCIENCES
DATA INFRASTRUCTURE

Python (II) - Programs - Iteration

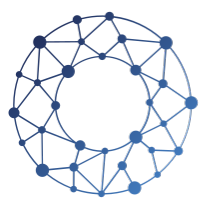
- iteratively multiply a number `n` by `m`
- iteratively find if a string `s` is a palindrome
- iteratively find the `n`th Fibonacci number



PSDI
PHYSICAL SCIENCES
DATA INFRASTRUCTURE

Python (II) - Programs - Iteration

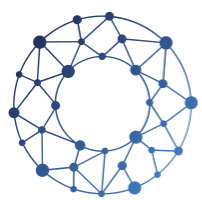
- iteratively multiply a number `n` by `m`
- iteratively find if a string `s` is a palindrome
- iteratively find the `n`th Fibonacci number
- iteratively find `n` factorial



PSDI
PHYSICAL SCIENCES
DATA INFRASTRUCTURE

Python (II) - Programs - Iteration

- iteratively multiply a number `n` by `m`
- iteratively find if a string `s` is a palindrome
- iteratively find the `n`th Fibonacci number
- iteratively find `n` factorial

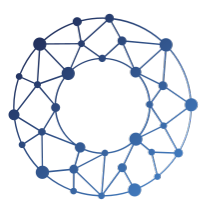


PSDI
PHYSICAL SCIENCES
DATA INFRASTRUCTURE

Python (II) - Programs - Recursion

Recursion.. works by induction

- start with base case
- recur to get to base case

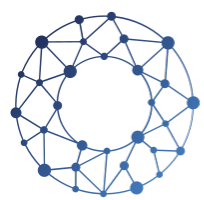


PSDI
PHYSICAL SCIENCES
DATA INFRASTRUCTURE

Python (II) - Programs - Recursion

This is when you can use a function to solve itself. To do:

- recursively multiply a number `n` by `m`

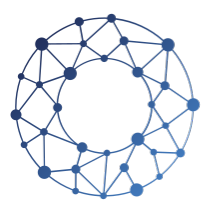


PSDI
PHYSICAL SCIENCES
DATA INFRASTRUCTURE

Python (II) - Programs - Recursion

This is when you can use a function to solve itself. To do:

- recursively multiply a number `n` by `m`
- recursively find if a string `s` is a palindrome

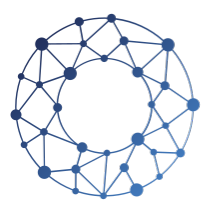


PSDI
PHYSICAL SCIENCES
DATA INFRASTRUCTURE

Python (II) - Programs - Recursion

This is when you can use a function to solve itself. To do:

- recursively multiply a number `n` by `m`
- recursively find if a string `s` is a palindrome
- recursively find the `n`th Fibonacci number

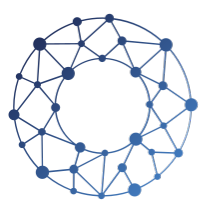


PSDI
PHYSICAL SCIENCES
DATA INFRASTRUCTURE

Python (II) - Programs - Recursion

This is when you can use a function to solve itself. To do:

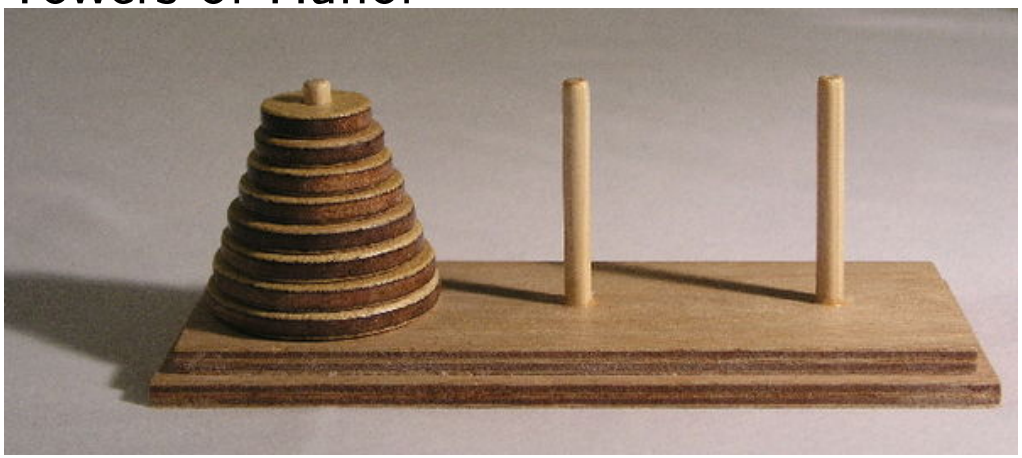
- recursively multiply a number `n` by `m`
- recursively find if a string `s` is a palindrome
- recursively find the `n`th Fibonacci number
- recursively find an element in a sorted list



PSDI
PHYSICAL SCIENCES
DATA INFRASTRUCTURE

Python (II) - Programs - Recursion

Towers of Hanoi



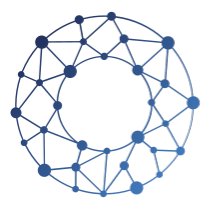
- Invented by French mathematician Édouard Lucas 1883
- you have a number of discs of descending size all on top of one another on a peg
- you need to move them to another peg
- you can only move one at a time
- you can only have smaller rings on top of larger rings

How do you move n rings from one pole to another?

PSDI
PHYSICAL SCIENCES DATA INFRASTRUCTURE

Python (II) - Programs - Recursion

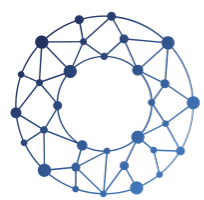
Factorial
Memoisation - if time



PSDI
PHYSICAL SCIENCES
DATA INFRASTRUCTURE

Python (II) - Programs - Lambda Functions

Anonymous functions that are simple and useful. can use in filter, or to return other functions.
have form `lambda x:x**2`



PSDI
PHYSICAL SCIENCES
DATA INFRASTRUCTURE