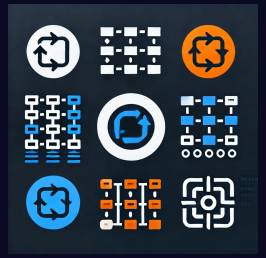


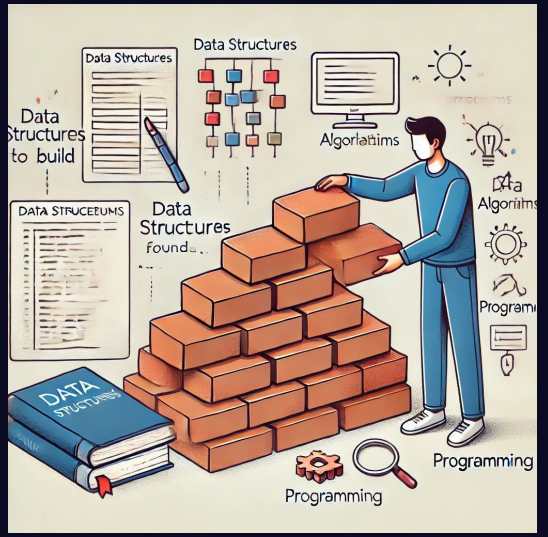
Welcome to Algorismo!

Unlock the World of Algorithms and Coding

Purpose of the Series

Purpose of the Series

Purpose of the Series



What We Will Cover

- Start with basic tasks:
  Searching for a number, **sorting** names

What We Will Cover

- Start with basic tasks:
  Searching for a number, **sorting** names

- Move to more **complex organization**:
  To-do lists, file organization

What We Will Cover

- Start with basic tasks:
  Searching for a number, **sorting** names

- Move to more **complex organization**:
  To-do lists, file organization

- Tackle **advanced problems**: City maps, weather forecast

What is an Algorithm?

- An algorithm is a **step-by-step** procedure for solving a problem.
  - Think of it as a **recipe** for your computer.

What is an Algorithm?

- An algorithm is a **step-by-step** procedure for solving a problem.
  - Think of it as a **recipe** for your computer.

What is an Algorithm?

- An algorithm is a **step-by-step** procedure for solving a problem.
  - Think of it as a **recipe** for your computer.

What is an Algorithm?

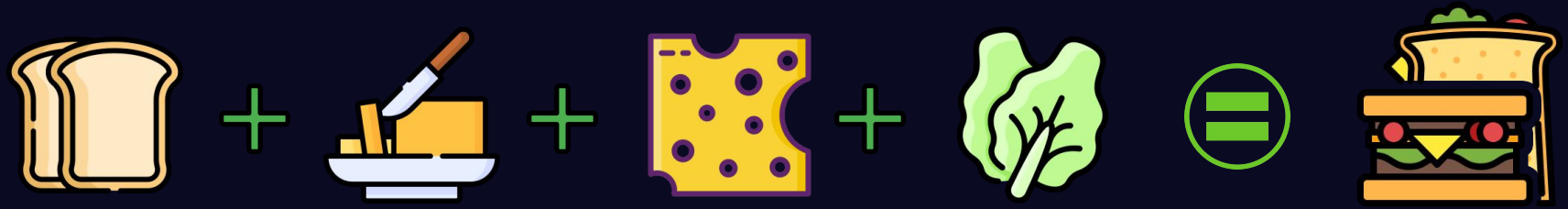
- An algorithm is a **step-by-step** procedure for solving a problem.
  - Think of it as a **recipe** for your computer.

What is an Algorithm?

- An algorithm is a **step-by-step** procedure for solving a problem.
  - Think of it as a **recipe** for your computer.

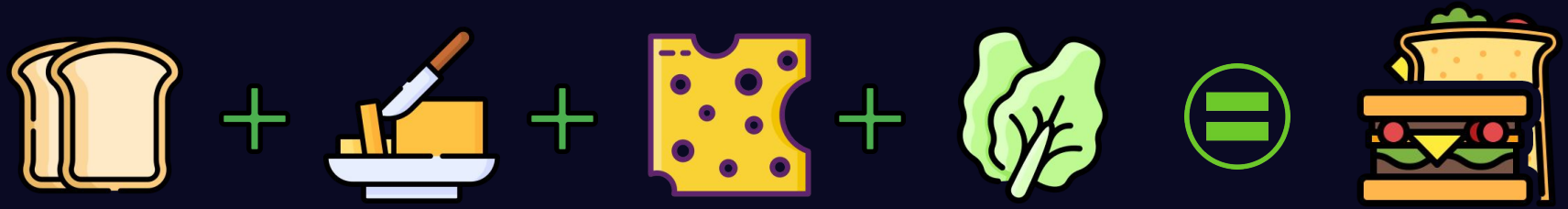
What is an Algorithm?

- An algorithm is a **step-by-step** procedure for solving a problem.
  - Think of it as a **recipe** for your computer.

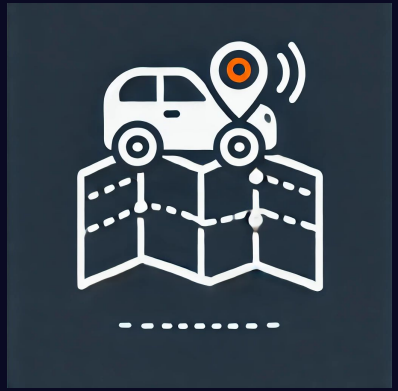


What is an Algorithm?

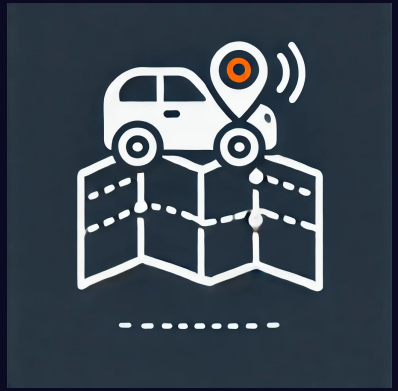
- An algorithm is a **step-by-step** procedure for solving a problem.
  - Think of it as a **recipe** for your computer.



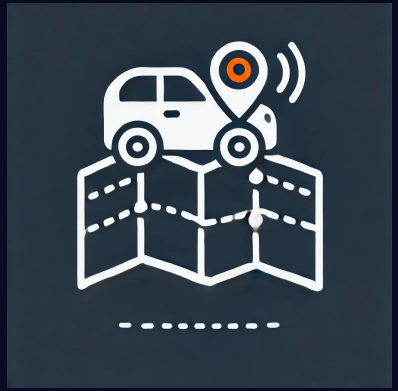
Why Algorithms Matter?



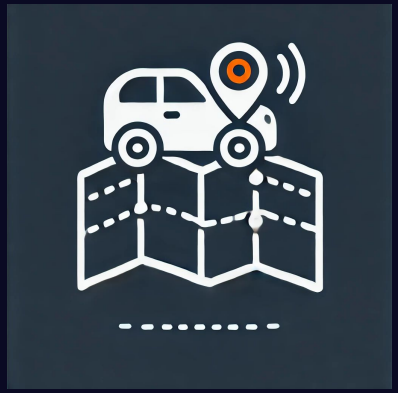
Why Algorithms Matter?



Why Algorithms Matter?



Why Algorithms Matter?



Key Traits of a Good Algorithm

- **Correct** – It should give you the right answer.

- **Efficient** – It should use resources wisely.

Key Traits of a Good Algorithm

- **Correct** – It should give you the right answer.

- **Efficient** – It should use resources wisely.

- **Finite** – It should end.

- **Clear** – It’s easy to understand.

- **General** – It solves a wide range of problems.

Basic Components of Algorithms

Basic Components of Algorithms

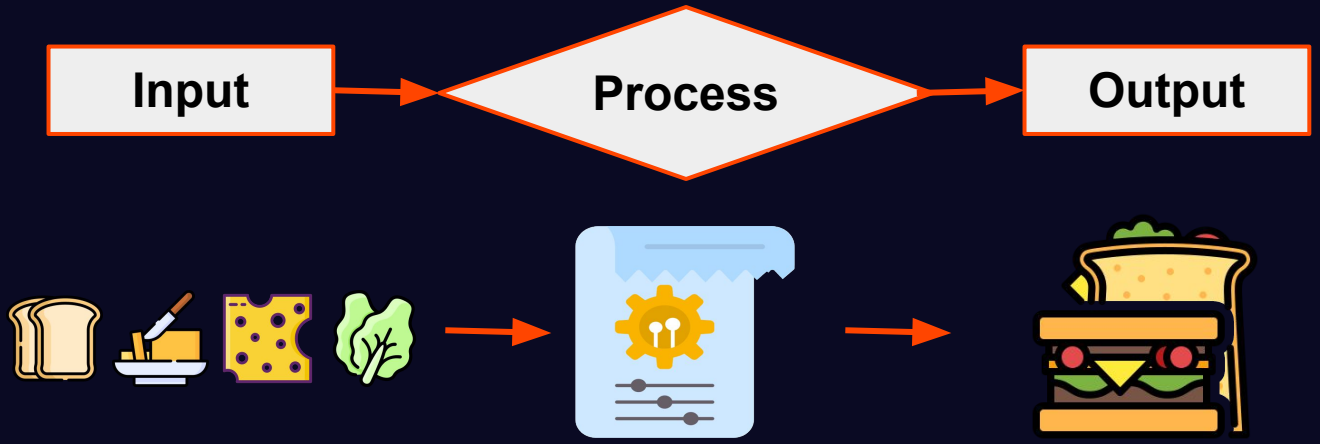
Input

Output

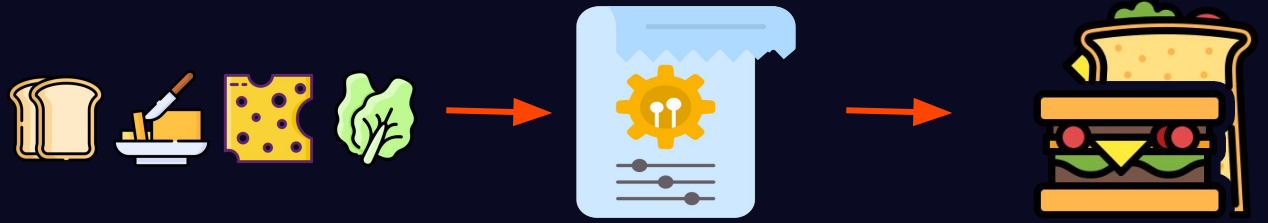
Basic Components of Algorithms

Basic Components of Algorithms

Basic Components of Algorithms

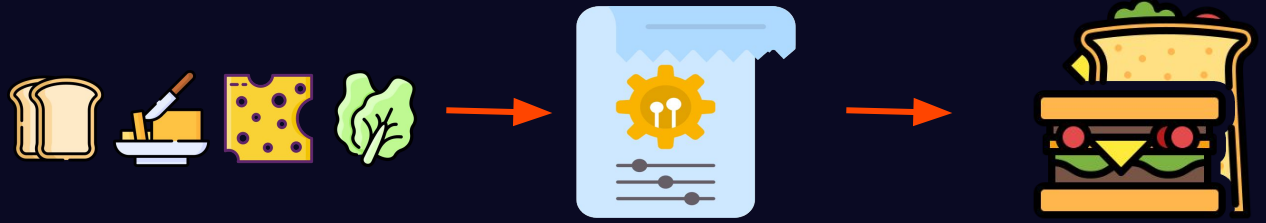


Basic Components of Algorithms

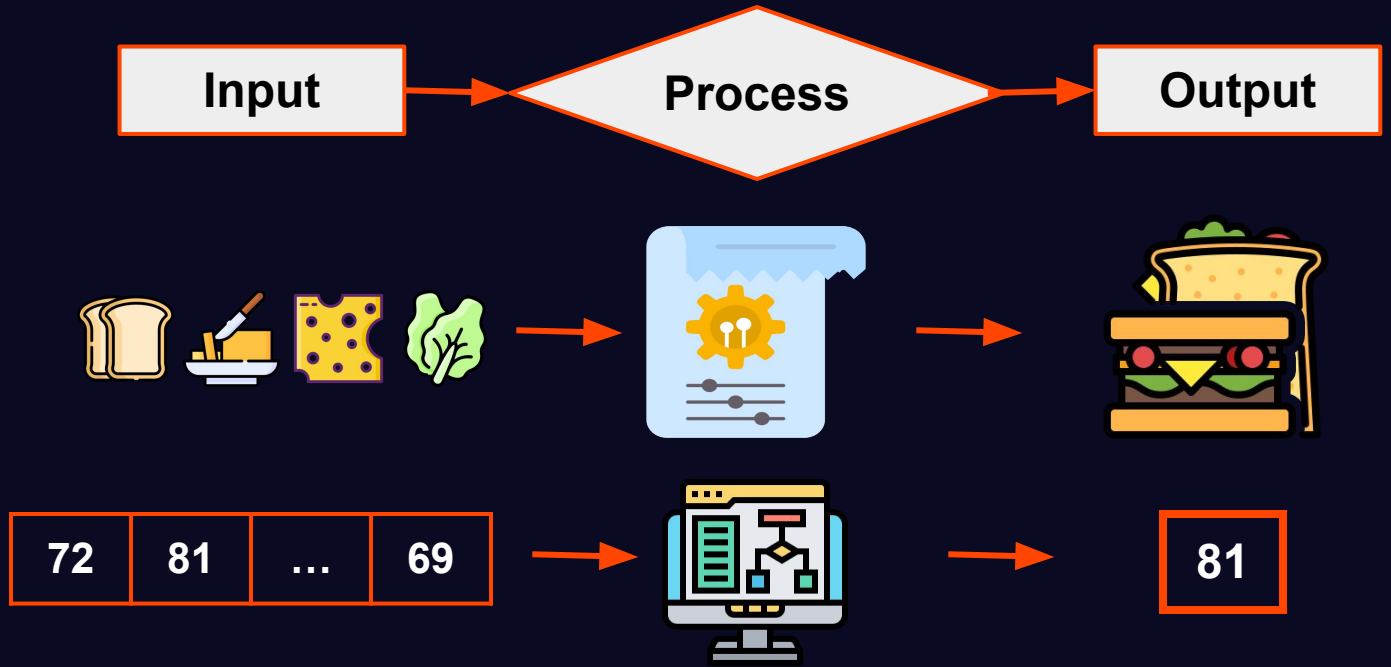


72	81	...	69

Basic Components of Algorithms



Basic Components of Algorithms



Algorithm in Action: Finding the Max

Input

Algorithm in Action: Finding the Max

Input

Output

Algorithm in Action: Finding the Max

Input

Output

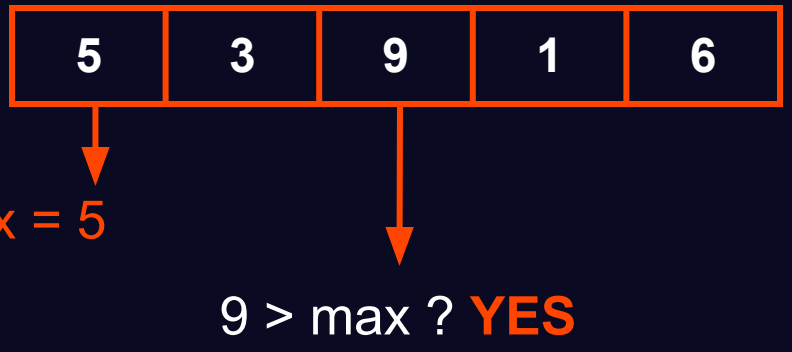
Algorithm in Action: Finding the Max

Input

Output

Algorithm in Action: Finding the Max

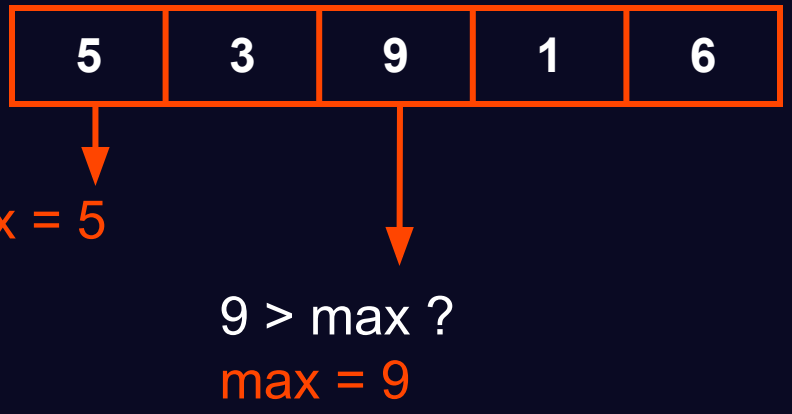
Input



Output

Algorithm in Action: Finding the Max

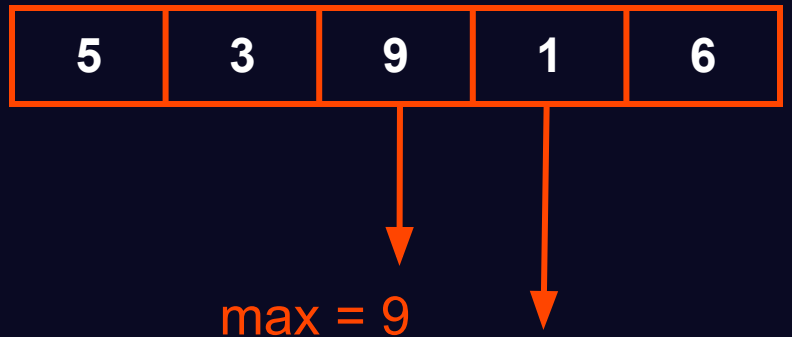
Input



Output

Algorithm in Action: Finding the Max

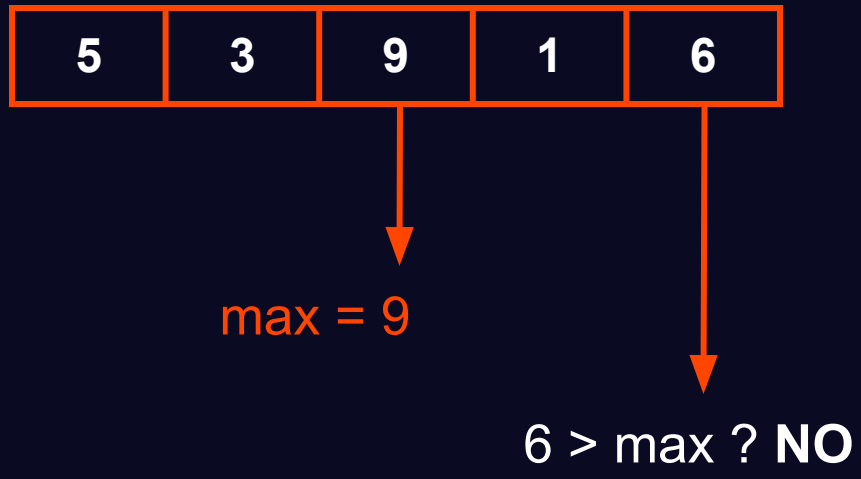
Input



Output

Algorithm in Action: Finding the Max

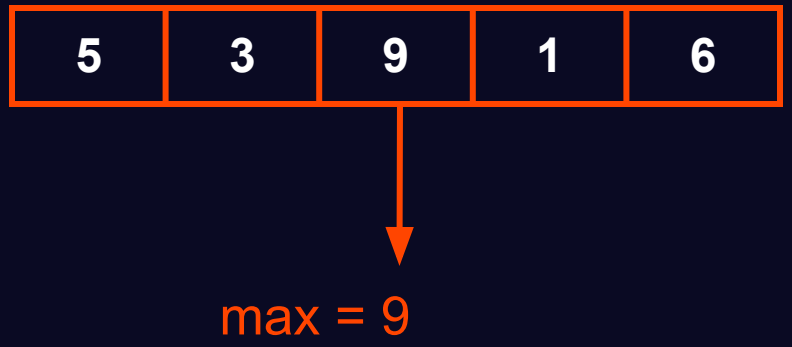
Input



Output

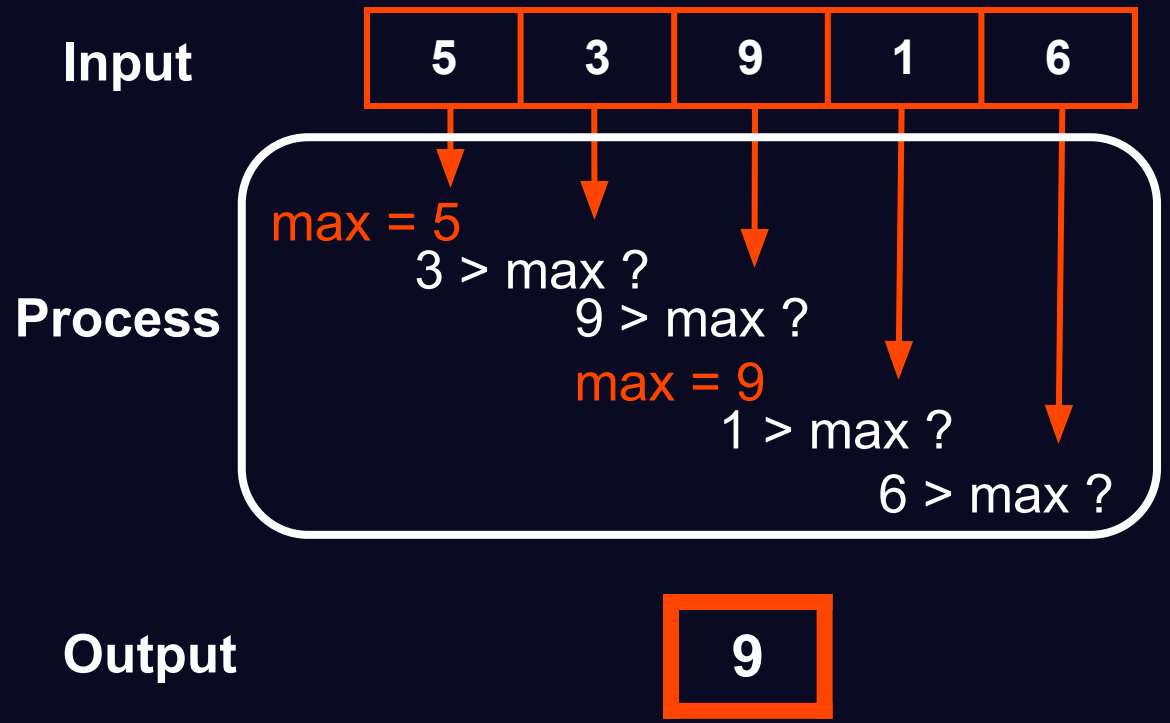
Algorithm in Action: Finding the Max

Input

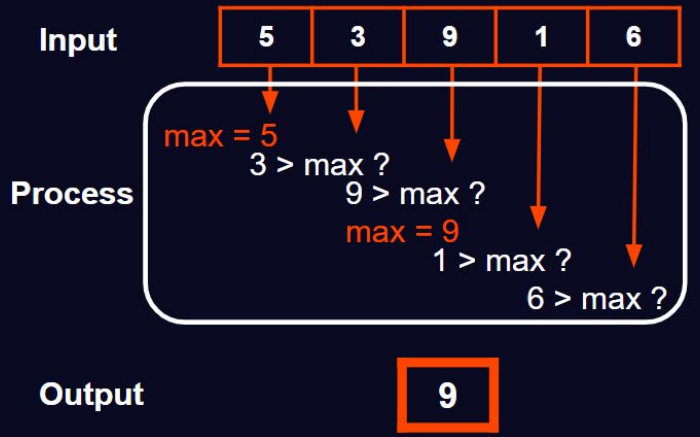


Output

Algorithm in Action: Finding the Max



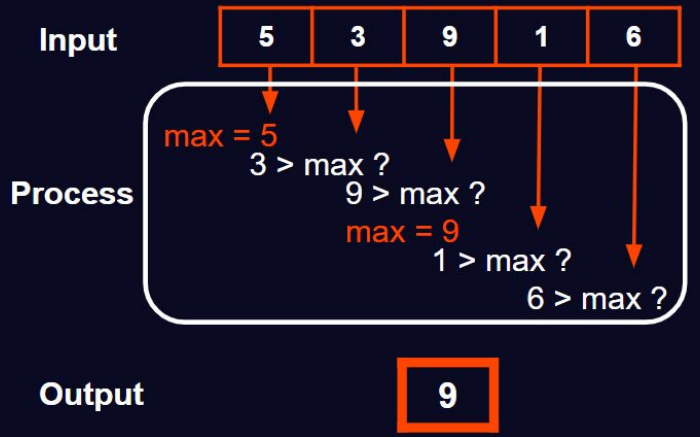
Describing Algorithm: Pseudocode



Describing Algorithm: Pseudocode

Step1:

Start with the first number, call it max.



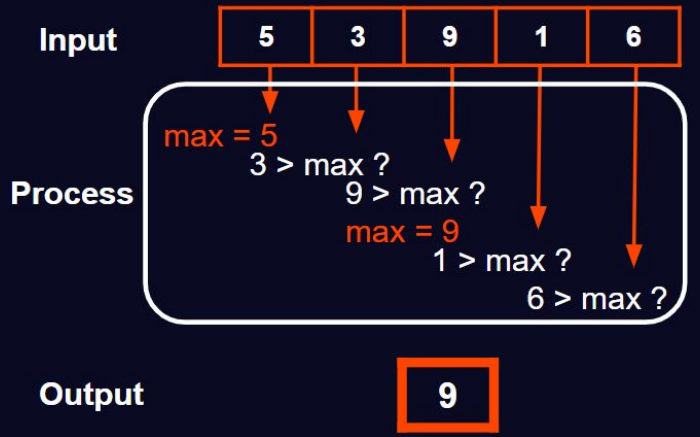
Describing Algorithm: Pseudocode

Step1:

Start with the first number, call it max.

Step2:

Compare it with each number in the list.



Describing Algorithm: Pseudocode

Step1:

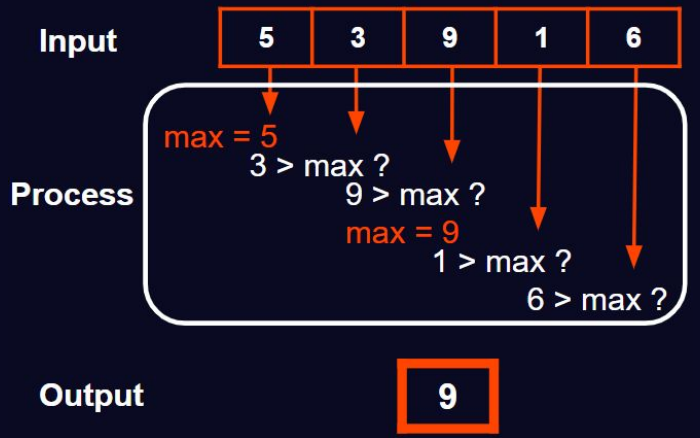
Start with the first number, call it max.

Step2:

Compare it with each number in the list.

Step3:

If you find a bigger number, update max.



Describing Algorithm: Pseudocode

Step1:

Start with the first number, call it max.

Step2:

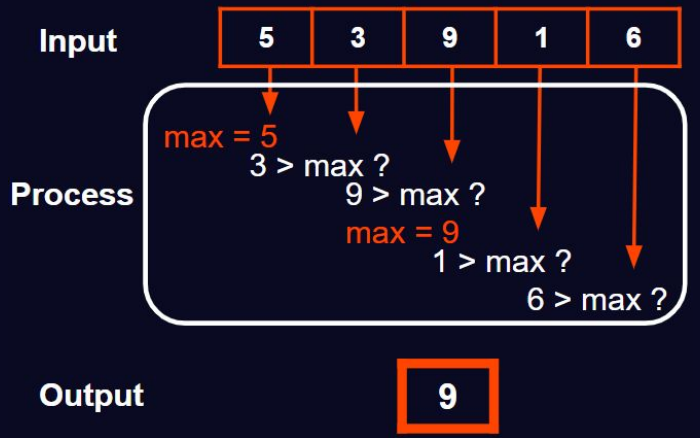
Compare it with each number in the list.

Step3:

If you find a bigger number, update max.

Step4:

At the end, max is the biggest number.



Formalizing Algorithm: Pseudocode

Formalizing Algorithm: Pseudocode

Algorithm FindMax
Input: A list of numbers
Output: The maximum number in the list

Formalizing Algorithm: Pseudocode

Algorithm FindMax
Input: A list of numbers
Output: The maximum number in the list

Begin

Step1: Start with the first number, call it max.

Formalizing Algorithm: Pseudocode

Algorithm FindMax
Input: A list of numbers
Output: The maximum number in the list

```
Begin
    max ← list[0]
    for each number in list do
        if number > max then
            max ← number
        end if
    end for
```

Step1: Start with the first number, call it max.

Step2: Compare it with each number in the list.

Step3: If you find a bigger number, update max.

Formalizing Algorithm: Pseudocode

Algorithm FindMax
Input: A list of numbers
Output: The maximum number in the list

```
Begin
    max ← list[0]

    for each number in list do
        if number > max then
            max ← number
        end if
    end for
    return max
End
```

Step1: Start with the first number, call it max.

Step2: Compare it with each number in the list.

Step3: If you find a bigger number, update max.

Step4: At the end, max is the biggest number.

Implementing Algorithm in Python

```
numbers = [5, 3, 9, 1, 6]
```

Implementing Algorithm in Python

numbers = [5, 3, 9, 1, 6]

```
max_number = numbers[0]
```

Step1: Start with the first number, call it max_number.

Implementing Algorithm in Python

```
numbers = [5, 3, 9, 1, 6]

max_number = numbers[0]

for number in numbers:
    if number > max_number:
        max_number = number
```

Step1: Start with the first number, call it max_number.

Step2: Compare it with each number in the list.

Step3: If you find a bigger number, update max_number.

Implementing Algorithm in Python

```
numbers = [5, 3, 9, 1, 6]

max_number = numbers[0]

for number in numbers:
    if number > max_number:
        max_number = number
```

Step1: Start with the first number, call it max_number.

Step2: Compare it with each number in the list.

Step3: If you find a bigger number, update max_number.

Implementing Algorithm in Python

```
numbers = [5, 3, 9, 1, 6]

max_number = numbers[0]

for number in numbers:
    if number > max_number:
        max_number = number

print("The maximum number is:", max_number)
```

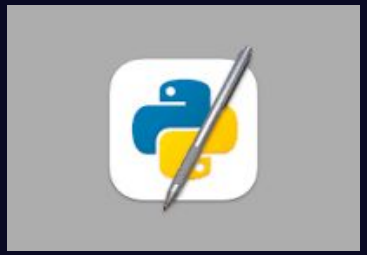
Step1: Start with the first number, call it max_number.

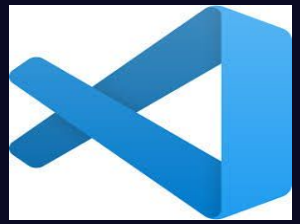
Step2: Compare it with each number in the list.

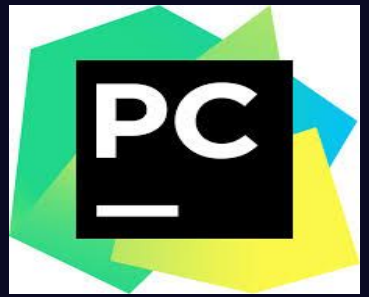
Step3: If you find a bigger number, update max_number.

Step4: At the end, max_number is the biggest number.

Running our Python Code







Running our Python Code

https://pythontutor.com/python-compiler.html

Converting Algorithm to Function

Converting Algorithm to Function

```
numbers = [5, 3, 9, 1, 6]
max_number = numbers[0]
for number in numbers:
    if number > max_number:
        max_number = number
print("The maximum number is:", max_number)
```

Converting Algorithm to Function

~~numbers = [5, 3, 9, 1, 6]~~

```
max_number = numbers[0]
for number in numbers:
    if number > max_number:
        max_number = number
```

~~print("The maximum number is:", max_number)~~

Converting Algorithm to Function

```
numbers = [5, 3, 9, 1, 6]
max_number = numbers[0]
for number in numbers:
    if number > max_number:
        max_number = number
print("The maximum number is:", max_number)
```

```
sample_numbers = [5, 3, 9, 1, 6]
print("The maximum number is:", max_number)
```

Converting Algorithm to Function

```
def find_max(numbers):

    max_number = numbers[0]
    for number in numbers:
        if number > max_number:
            max_number = number


sample_numbers = [5, 3, 9, 1, 6]
print("The maximum number is:", max_number)
```

Converting Algorithm to Function

```
def find_max(numbers):
```

Indented Block

```
    max_number = numbers[0]
    for number in numbers:
        if number > max_number:
            max_number = number
```

```
sample_numbers = [5, 3, 9, 1, 6]
print("The maximum number is:", max_number)
```

Converting Algorithm to Function

```
def find_max(numbers):
    max_number = numbers[0]
    for number in numbers:
        if number > max_number:
            max_number = number
    return max_number

sample_numbers = [5, 3, 9, 1, 6]
print("The maximum number is:", max_number)
```

Indented Block

Converting Algorithm to Function

```
def find_max(numbers):
    max_number = numbers[0]
    for number in numbers:
        if number > max_number:
            max_number = number
    return max_number

sample_numbers = [5, 3, 9, 1, 6]
print("The maximum number is:", find_max(sample_number))
```

Indented Block

Converting Algorithm to Function

```
def find_max(numbers):
    max_number = numbers[0]
    for number in numbers:
        if number > max_number:
            max_number = number
    return max_number

sample_numbers = [5, 3, 9, 1, 6]
print("The maximum number is:", find_max(sample_number))
```

Indented Block

Converting Algorithm to Function

```
def find_max(numbers):
    max_number = numbers[0]
    for number in numbers:
        if number > max_number:
            max_number = number
    return max_number

sample_numbers = [5, 3, 9, 1, 6]
print("The maximum number is:", find_max(sample_number))
```

Parameters

Indented Block

Arguments

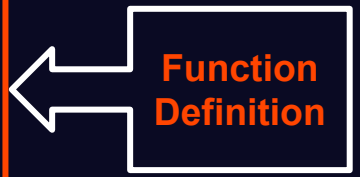
Converting Algorithm to Function

```
def find_max(numbers):

    max_number = numbers[0]
    for number in numbers:
        if number > max_number:
            max_number = number
    return max_number

sample_numbers = [5, 3, 9, 1, 6]
print("The maximum number is:", find_max(sample_number))
```

Indented Block

Converting Algorithm to Function

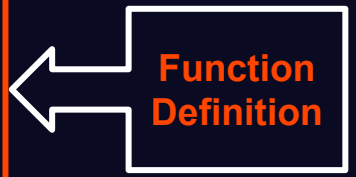
```
def find_max(numbers):
    max_number = numbers[0]
    for number in numbers:
        if number > max_number:
            max_number = number
    return max_number
```




```
sample_numbers = [5, 3, 9, 1, 6]
print("The maximum number is:", find_max(sample_number))
```

Converting Algorithm to Function

```
def find_max(numbers):
    max_number = numbers[0]
    for number in numbers:
        if number > max_number:
            max_number = number
    return max_number
```




```
sample_numbers = [5, 3, 9, 1, 6]
print("The maximum number is:", find_max(sample_number))
```

Summary and What’s Next

- Summary
  - Understanding Algorithms
  - Pseudocode
  - Python Implementation
  - Functions

Summary and What's Next

- Summary
  - Understanding Algorithms
  - Pseudocode
  - Python Implementation
  - Functions

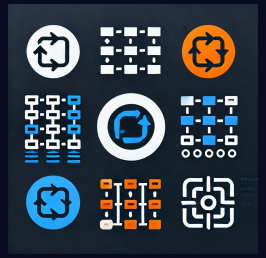
- Exercise
  - An exercise to count the steps it takes to find the maximum number.
  - Solution is provided
- Link to examples, exercises, and solutions
  - https://github.com/rmateeq/algorismo

Summary and What's Next

- Summary
  - Understanding Algorithms
  - Pseudocode
  - Python Implementation
  - Functions

- **Next Topic**
  - **Linear Search**

- Exercise
  - An exercise to count the steps it takes to find the maximum number.
  - Solution is provided
- Link to examples, exercises, and solutions
  - https://github.com/rmateeq/algorismo



Thank You for Watching!