Newton School of Technology

22

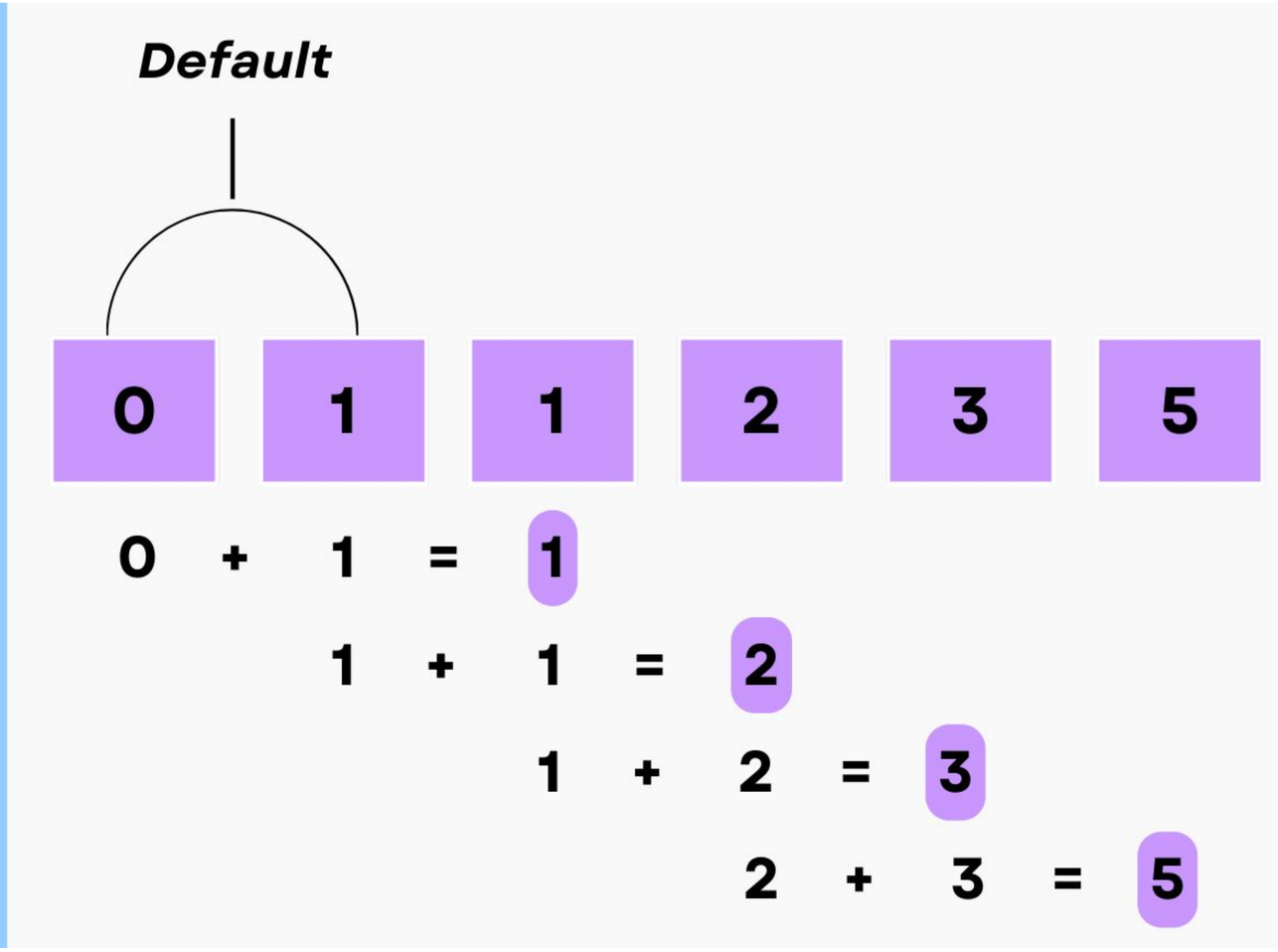
Recursion Part-2

by Gladden Rumao

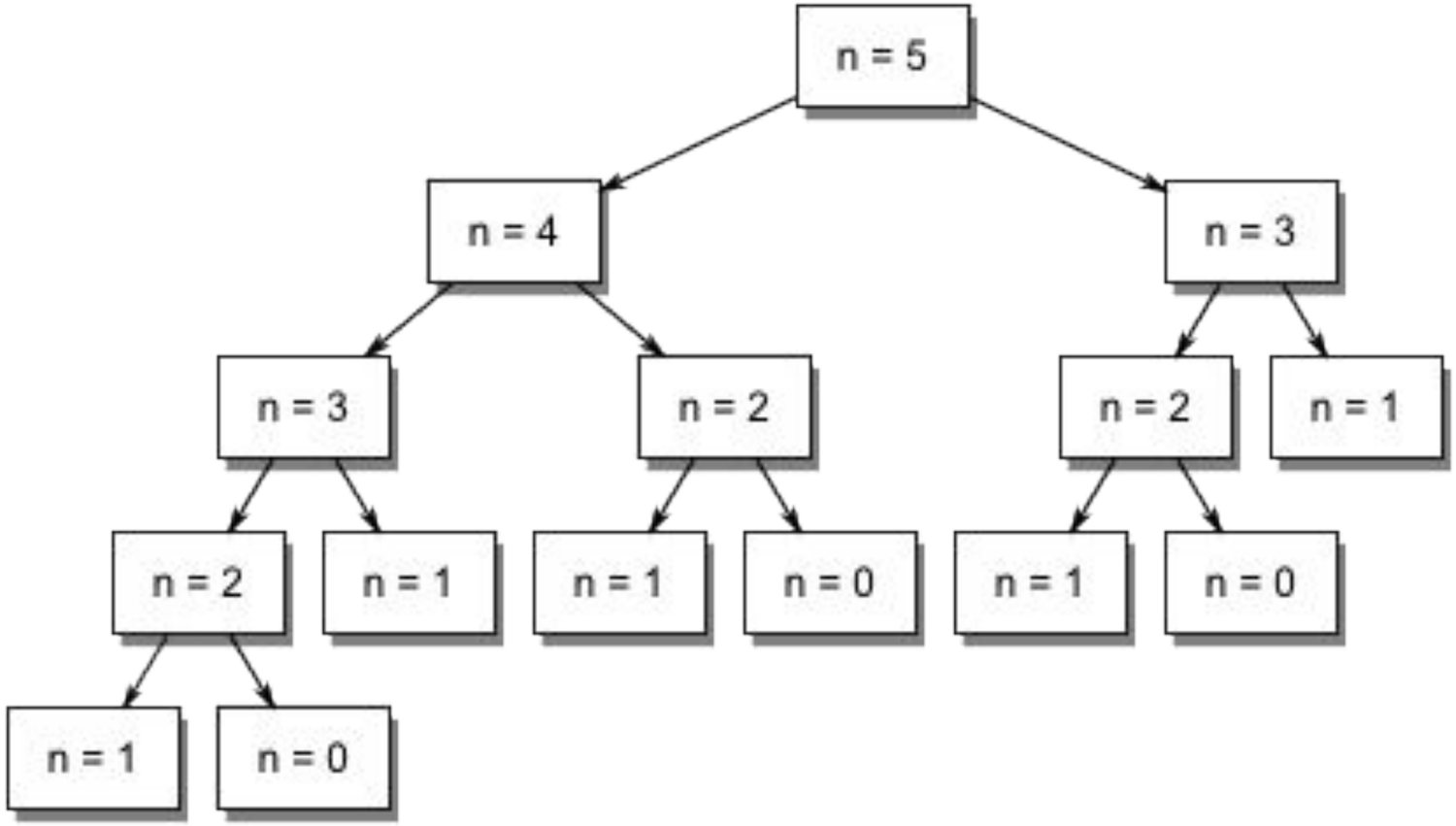
CSA101 : Problem Solving with Programming

Example- Fibonacci

Fibonacci Series



Example- Fibonacci



Example- Factorial Visualization

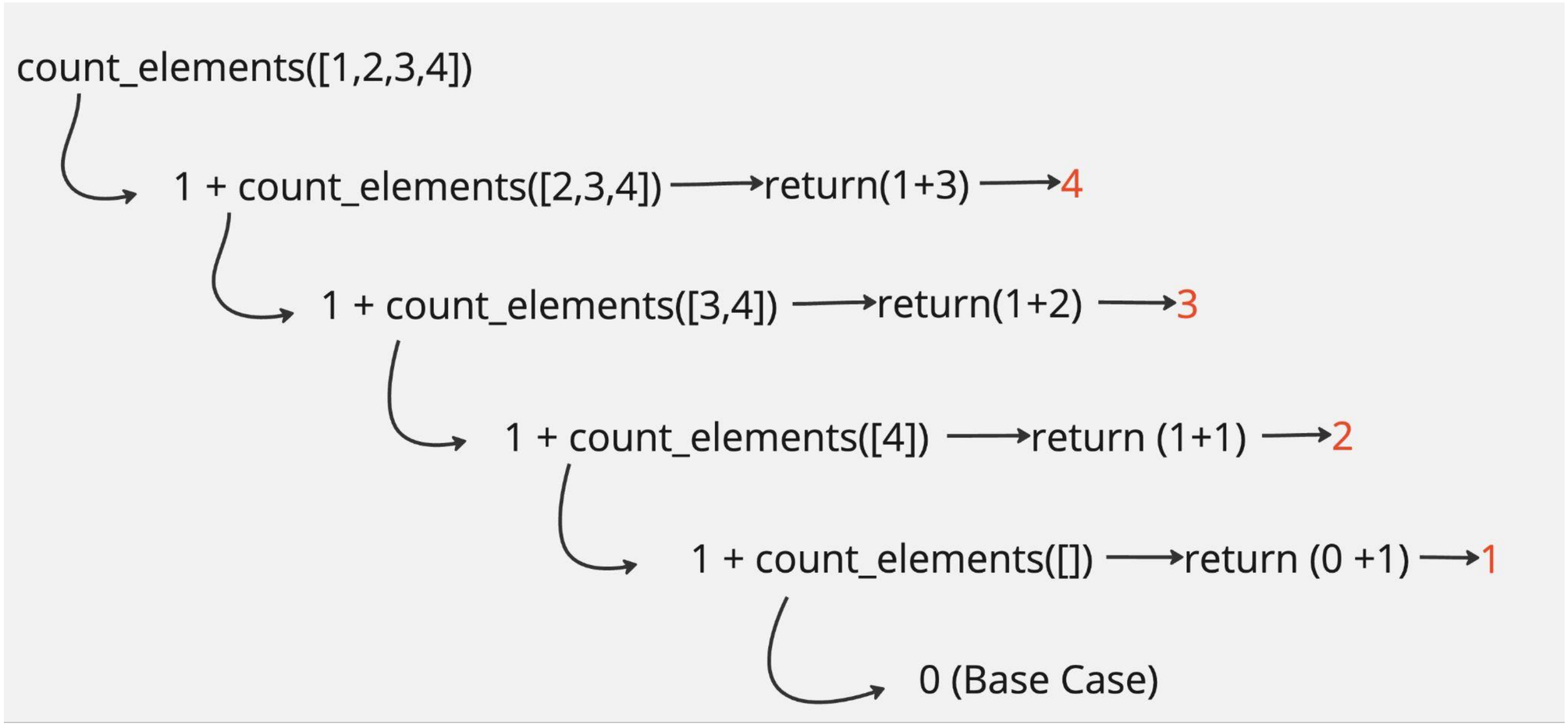
https://www.educative.io/courses/recursion-for-coding-interviews-in-java/recursion-and-memory-visualization

Recursion Mechanism

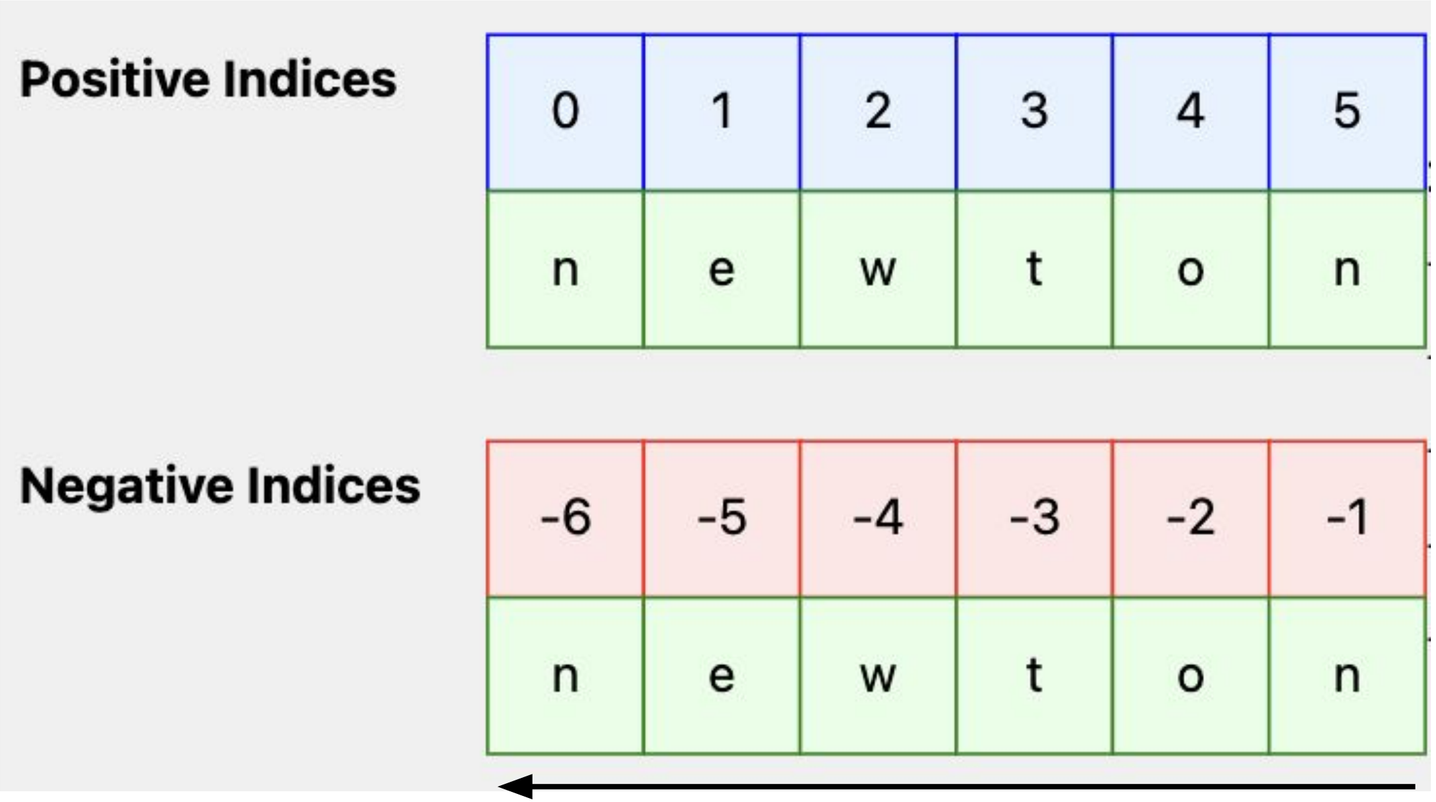
Newton School of Technology

[Online Python Tutor - Visualize program execution]

Question: Count elements in the List



Question: Reverse of String



Question: Palindrome

Newton School of Technology

Level	String	s[0]	s[-1]	Check	Result	Recursive Call
1	“racecar”	‘r’	‘r’	‘r’ == ‘r’	Recur ‘aceca’	is_palindrome(‘aceca’)
2	“aceca”	‘a’	‘a’	‘a’ == ‘a’	Recur ‘cec’	is_palindrome(‘cec’)
3	“cec”	‘c’	‘c’	‘c’ == ‘c’	Recur ‘e’	is_palindrome(‘e’)
4	“e”	‘e’	‘e’	Base Case	True	-

When to use Recursion

- When a problem can be broken down into smaller, similar subproblems.
- When calculating values that can be defined in terms of themselves
  Example- factorial of a number (n!)
- When dealing with data that has multiple layers of nesting, such as lists within lists or objects within objects.
- When generating combinations, permutations, or subsets of a set.

Advantages and Disadvantages

Newton School of Technology

Advantages:

Simplified Code

Natural Fit for Certain Problems

Ease of Implementation

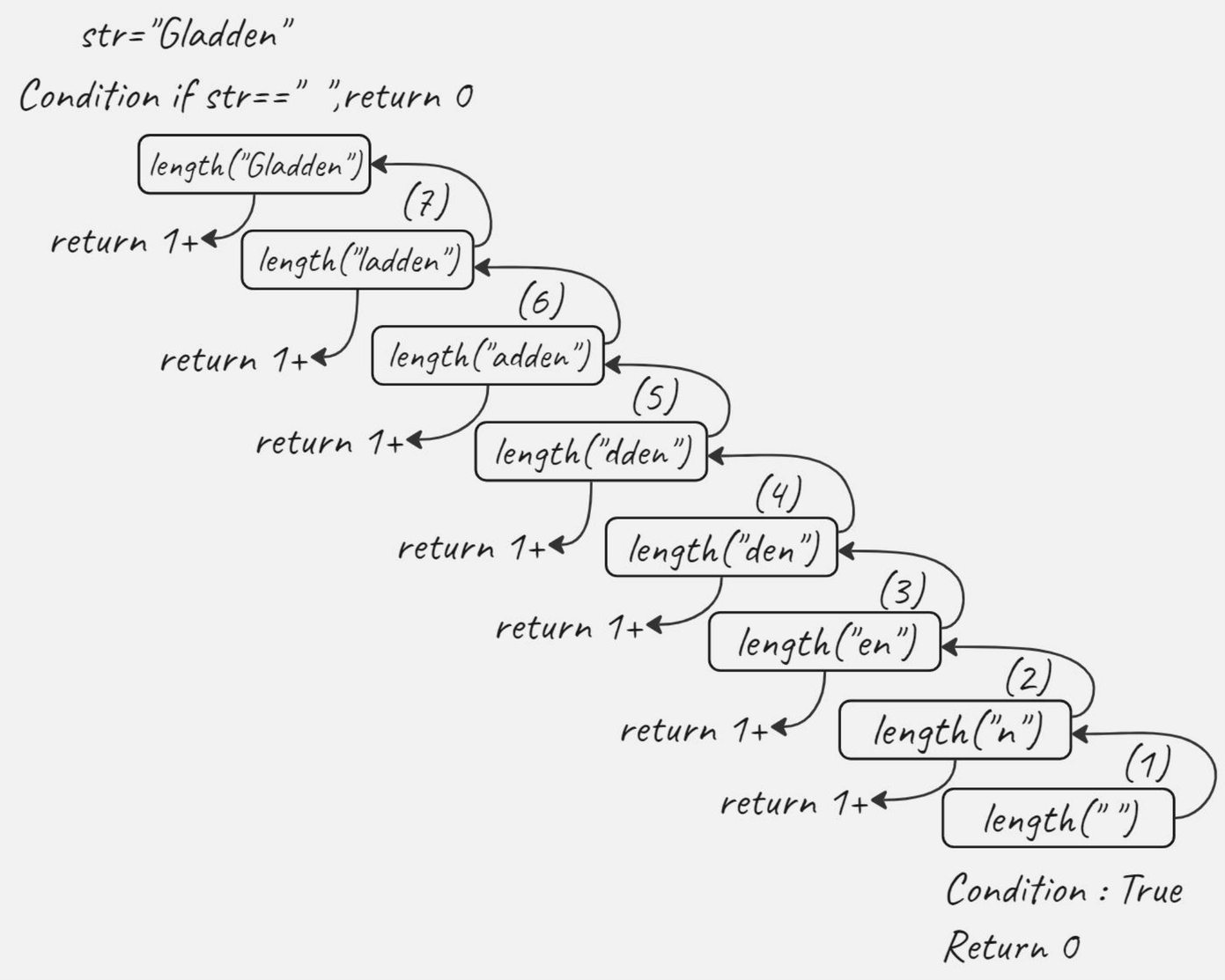
Disadvantages:

Performance Issues

Higher Memory Usage:

Base Case Complexity

Question: Length of a string



Question: Length of a string

```python
def length(s):
    if s == "":
        return 0
    return 1 + length(s[1:])

string = input("Enter the string: ")
print("Length of", string, "is", length(string))
```

Output

```
Enter the string: Gladden
Length of Gladden is 7
```

Thank You!