# Prologue

This book introduces many mathematical and computer science terms like Fobonacci, graph, tree, recursion, etc. While these concepts all have rigid formal definitions, they are not so easy to grasp in a glance. Instead, this book will try to describe them in simple words as much as possible.

Coming up with the right names for the characters in this book is equally important as making the algorithms and programs right. The names reflect our inspiration. Spending many nights staring at stars without a solution. But wait, since the computer world is an Animal Kingdom and our characters live in this Kingdom, therefore, we will name them -

Characters
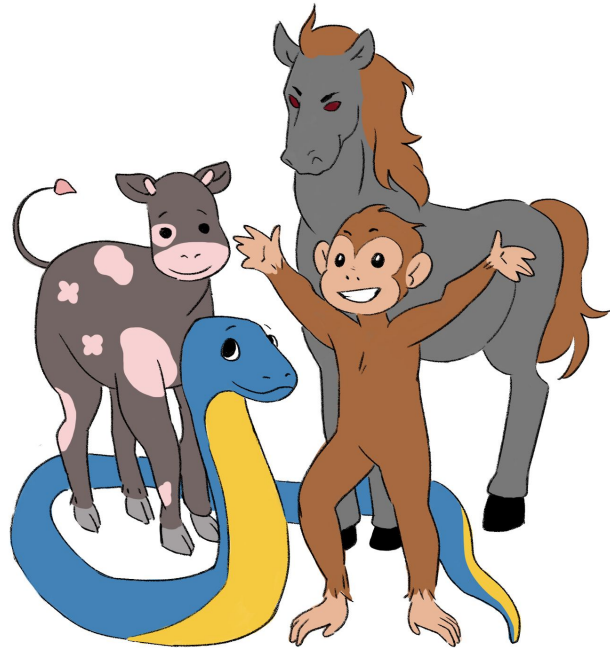


The sky is blue, the grass is green. deep in the forest, lives the BestFour:

- Dark Knight the Horse (male, college student)
    - Gallant hero and adventurer, independent and unrestrained, earn admiration from others, leader
- Banana Split the Monkey (male, 11th grader, high intelligent)
    - intelligent, curious, enthusiastic, confident, humorous, and sociable, can be moody and impatient at times
- Bubble Gum the Cow (girl, 6th grader)
    - Creative, good memory, cheerful, good-natured, seek peace and kindness, can settle for less at times

- Mighty Python the Algorithm
    - Artificial intelligent, a character to introduce algorithms and code
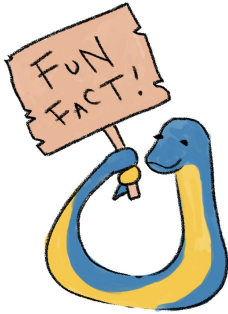


Computer World is an Animal Kingdom

Many programming languages and tools are named after animal names:
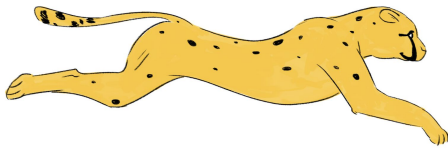


From as big as Elk (analytics software) to as tiny as Ant (build automation tool)



From friendly social Lama (heterogeneous system development) to deadly and evil Cobra (OOP language)

From as sleek as silk Penguin (graphical user interfaces development) to as sharp as Porcupine (web app server)



From as fast as Cheetah (template engine) to as slow as Sloth (slowest language by Larry Page)
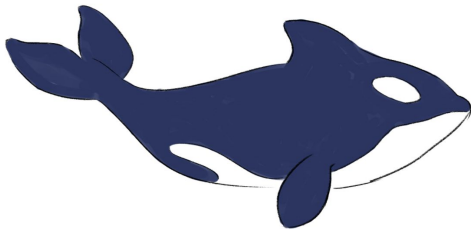


From as elegant as Orca (parallel application development) to as gross as Cockroach (SQL Database)

And sometimes, programmers just can not have enough of animals and add a whole family pack like Mac OS big cat family: 10.0 "Cheetah", 10.1 "Puma", 10.2 "Jaguar" ...

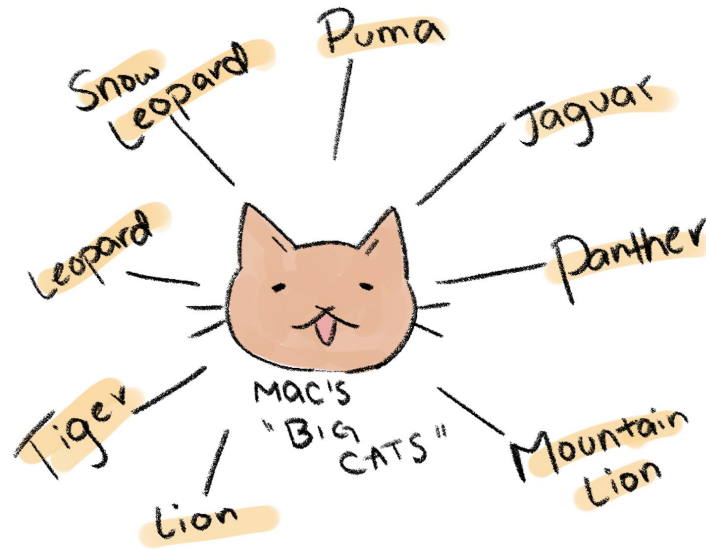Mac's "BIG CATS": Snow Leopard, Puma, Jaguar, Panther, Leopard, Mountain Lion, Tiger, Lion

You get the point!

Fun Fact - is Python language named after snake?

When Guido van Rossum began implementing Python in the 1980s, he was reading the comedy series "Monty Python's Flying Circus". Van Rossum thought he needed a name that was short, unique, and slightly mysterious, so he decided to call the language Python. The language was first released in 1991.

https://docs.python.org/2/faq/general.html#why-is-it-called-python

## Why Python?

Python is an easy-going friend to make. If you know English language, you've known a portion of Python! It resembles the English language and focuses on what you want to achieve, but not the programming language itself.

Python is a celebrity on all computer continents: Windows, MacOS, Linux. Your Python code works no matter where it runs.

Python is a big brain for you to pick, the artificial intelligence brain. With the support of Python clans (libraries) like **'Numpy', 'Pandas' and 'Matplotlib'**, Python handles statistics, matrix data and visualization very well. Whatsmore, some of the strong family members, "Keras", "TensorFlow" and "OpenCV" enable artificial intelligence. It is like you have two brains now.

## Target Audience

This book is designed to introduce the essential framework that computer science, math and even our lives are built upon: algorithmic thinking. Algorithm in simple words: the sequence of steps to solve a category of problems.
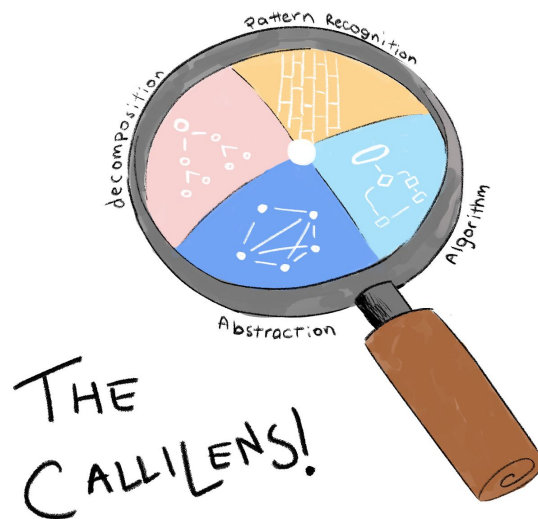


In this book, we will grant you a magic lens called CalliLens. With this lens, we see a crystal clear world. Algorithm sounds like a scary monster topic? Don't worry, we

introduced lots of visual illustrations and fun facts. If you like to watch movies and take adventurous journeys, you are in the right place. Welcome our adventure buddies!

## Knowledge Covered

### Algorithms

- Coin Change with Greedy Approach
- Fractional Knapsack Problem with Greedy Approach
- Coin Change with Dynamic Programming
- 0-1 Knapsack Problem with Dynamic Programming
- Shortest Path Problem with Minimum Spanning Tree Prim's and Kruskal's Algorithms (Greedy Approach)
- Single Source Shortest Path with Dijkstra's Algorithm (Greedy Approach)
- Solving Maze with Breadth-First Search and Depth-First Search

### Data Structures and Important Concepts

- Fibonacci
- Recursion
- Queue
- Stack
- Graph
- Tree
- etc.

Flow for Each Adventure

- Daily adventures

- CalliLens guides us through the abstraction, pattern recognition and solution process.

- Flow diagram. If the algorithm is the coffee, the flow diagram is the coffee mate!
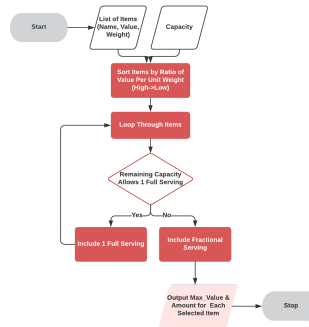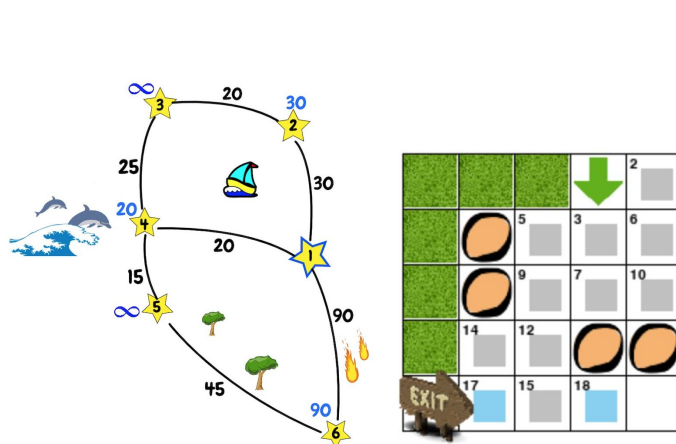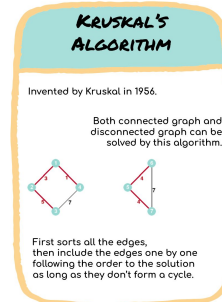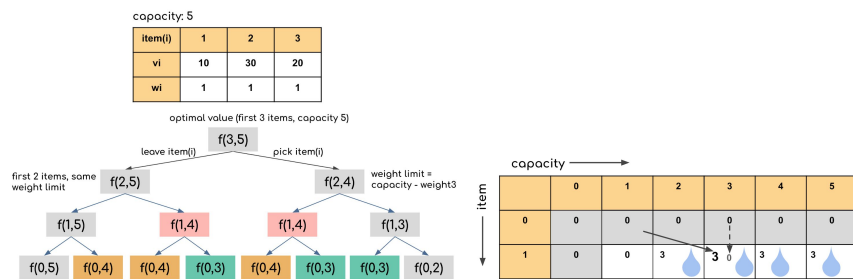


- Python solution and fun pygame

- Connecting to the real world

Throughout the book, we introduced lots of visual illustrations.

capacity: 5

| item(i) | 1 | 2 | 3 |
|---------|----|----|----|
| vi | 10 | 30 | 20 |
| wi | 1 | 1 | 1 |

optimal value (first 3 items, capacity 5)

f(3,5)

leave item(i)     pick item(i)

first 2 items, same weight limit   f(2,5)    f(2,4)   weight limit = capacity - weight3

f(1,5)   f(1,4)   f(1,4)   f(1,3)

f(0,5)   f(0,4)   f(0,4)   f(0,3)   f(0,4)   f(0,3)   f(0,3)   f(0,2)

capacity →

| item | | 0 | 1 | 2 | 3 | 4 | 5 |
|------|---|---|---|---|---|---|---|
| | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| | 1 | 0 | 0 | 3 | 0 | 3 | 3 |

**KRUSKAL'S ALGORITHM**

Invented by Kruskal in 1956.

Both connected graph and disconnected graph can be solved by this algorithm.

First sorts all the edges, then include the edges one by one following the order to the solution as long as they don't form a cycle.

1. Start exploring neighbours (LRUD)
Queue: (0,2) (0,4) (1,3)

therefore called Breadth First Search

2. Retrieve from Queue explore its neighbours
Queue: (0,4) (1,3) (0,1) (1,2)

3. Repeat step 2
Retrieve from Queue explore its neighbours
Queue: (1,3) (0,1) (1,2) (1,4)

4. When retrieving (1,1) from the Queue and exploring its neighbour, find EXIT

Mission accomplished!

BTW, do you have to love animals to continue reading this book? No, but it helps make a better world if you start now.

Oh, if you get to hunt a better solution than the one presented in the book, please share the joy with us!