Data Structures and Algorithms

2| Complexity, Graphs and Trees

Grégor JOUET
Phd Student
gregor.jouet@ext.devinci.fr

1

Introduction

(If you have questions, now would be a tremendous time to voice them)

2

Complexity

It’s not that complex

2 Complexity

How long ?

- How does the runtime of the function increase when you increase n ?

```python
def whaou(n):
  s = 0
  for i in range(n):
    s += i
  return s
```

```python
def whaou(n):
  s = 0
  for i in range(100*n):
    s += i
  return s
```

```python
def whaou(n):
  s = 0
  for i in range(n*n):
    s += i
  return s
```

```python
def whaou(n):
  s = 0
  for i in range(n*n*n):
    s += i
  return s
```

DE VINCI INNOVATION CENTER

It’s Math time !

- How do we express this difference in climb ?

$$f(x) = O(g(x))$$

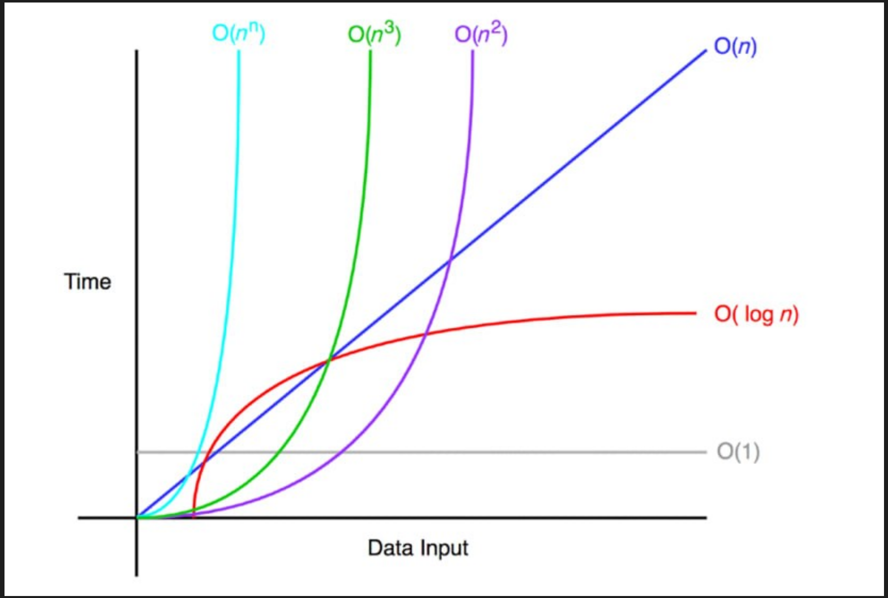
$$\lim_{x \to +\infty} \frac{f(x)}{g(x)} < \infty$$

$$f(x) = x^3 \; ; \; g(x) = 2^x$$

DE VINCI INNOVATION CENTER

How long ?

- Remember, we are only interested in the climb **rate**
- Computation time can grow rapidly !



DE VINCI INNOVATION CENTER

2 Complexity

How long ?

- You can see why time complexity can be interesting to know...

n / f(n)	log(n)	n	n log(n)	n^2	2^n	n!
10	0.003 µs	0.01 µs	0.033 µs	0.1 µs	1 µs	3.63 ms
20	0.004 µs	0.02 µs	0.086 µs	0.4 µs	1ms	77.1 years
30	0.005 µs	0.03 µs	0.147 µs	0.9 µs	1s	8.4×10^{15} years
40	0.005 µs	0.04 µs	0.213 µs	1.6 µs	18.3 min	
50	0.006 µs	0.05 µs	0.282 µs	2.5 µs	13 days.	
100	0.007 µs	0.1 µs	0.644 µs	10 µs	4×10^{13} years	
1,000	0.010 µs	1 µs	9.966 µs	1ms		
10,000	0.013 µs	10 µs	130 µs	100ms		
100,000	0.017 µs	0.10 ms	1.67 ms	10s		
1,000,000	0.020 µs	1ms	19.93 ms	16.7 min		
10,000,000	0.023 µs	0.01 s	0.23 s	1.16 days		
100,000,000	0.027 µs	0.1 s	2.66 s	115.7 days		
1,000,000,000	0.030 µs	1s	29.90 s	31.7 years		

DE VINCI INNOVATION CENTER

3

Graphs

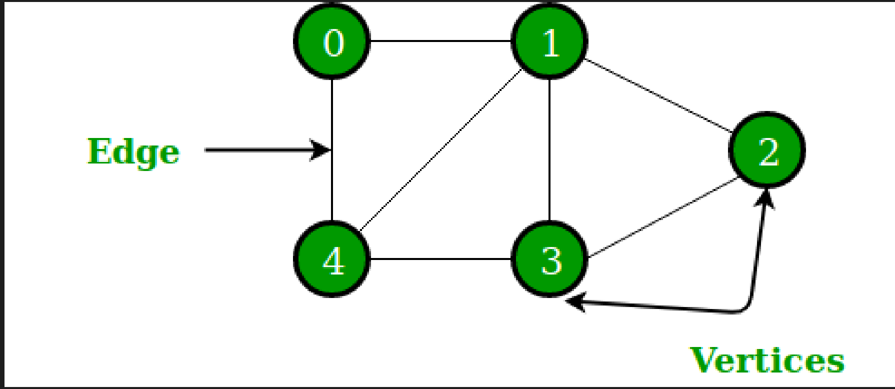
Let me draw that

DE VINCI INNOVATION CENTER

3 Graph

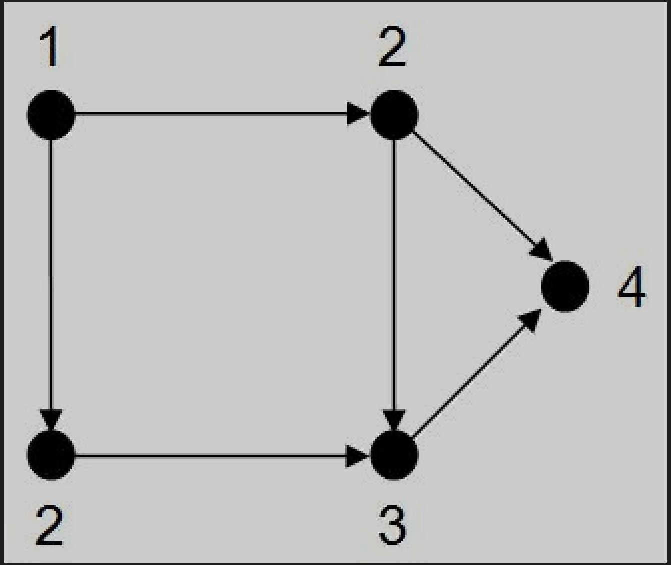
Definition

- Vertices (or nodes) and Edges
- Extremely wide range of applications
- A whole mathematical field dedicated to its study (Graph Theory)



Properties

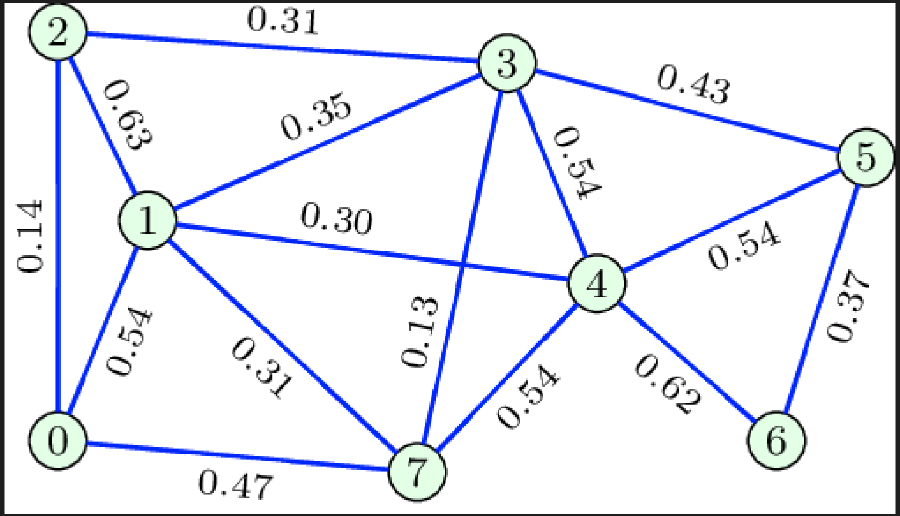
- Graphs can be oriented: You can go from 1 node to another



DE VINCI INNOVATION CENTER

Properties

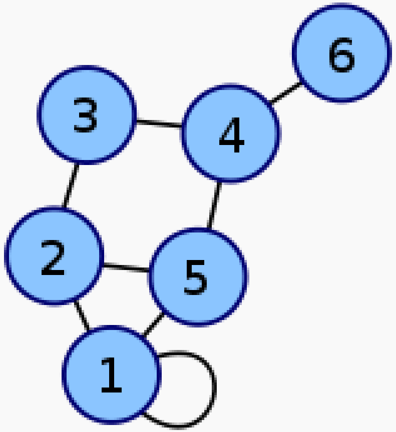
- Graphs can be **weighted**: Edges have an associated value



DE VINCI
INNOVATION
CENTER

Representation

- Graph can be represented with their **adjacency matrix**
- For undirected graphs, the matrix is symmetric

Labeled graph	Adjacency matrix
	$\begin{pmatrix} 1 & 1 & 0 & 0 & 1 & 0 \\ 1 & 0 & 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 1 & 1 \\ 1 & 1 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \end{pmatrix}$

4

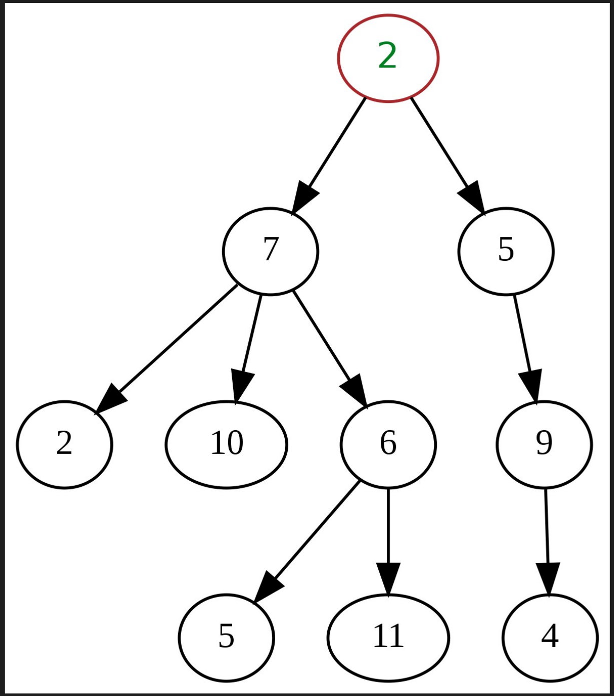
Trees

Growing data

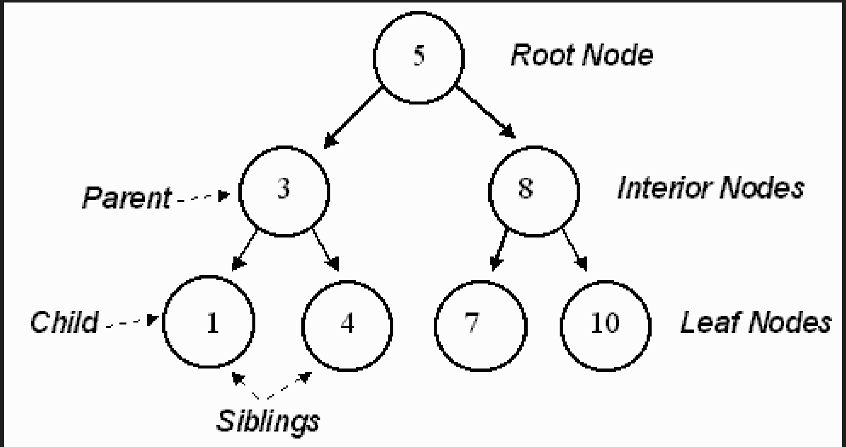
DE VINCI INNOVATION CENTER

What is it ?

- Graph in which any two vertices are connected by *exactly one* path.
- Trees are used in a variety of applications: file systems, compression, compilers, graphics, etc..

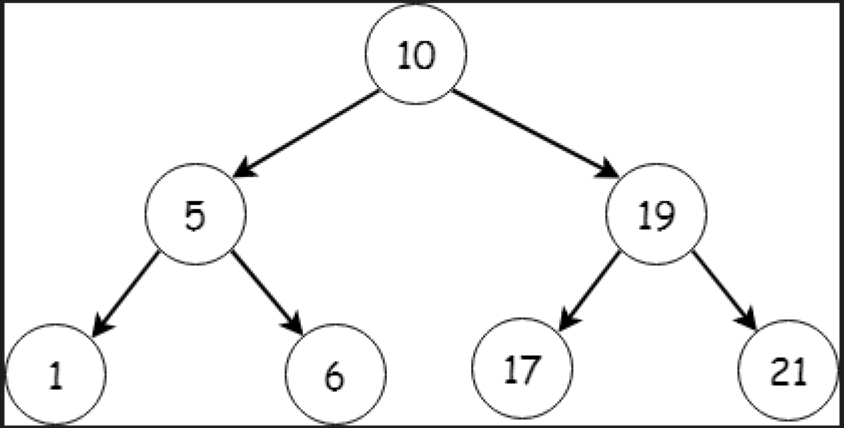


Vocabulary


5
Root Node
Parent
3
8
Interior Nodes
Child
1
4
7
10
Leaf Nodes
Siblings

Many flavors
B-Trees

- Binary trees: At most 2 children
- Used for Space Partition, Hash Maps, Databases



DE VINCI INNOVATION CENTER

Data Structures and Algorithms

3| Algorithmy & Algorithms

Grégor JOUET
Phd Student
gregor.jouet@ext.devinci.fr

1 Introduction

2 General method

3 Patterns

4

1

Introduction

Can’t write Algorithm without rhythm

DE VINCI INNOVATION CENTER

2

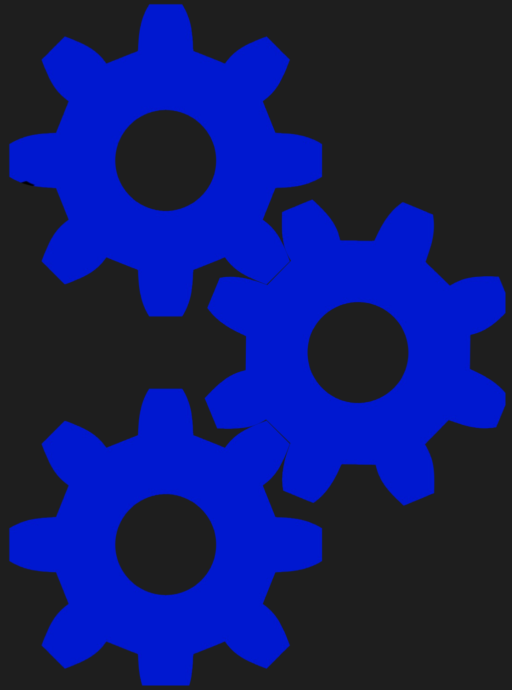
General Method

Read. Apply.

DE VINCI INNOVATION CENTER

General Method

- Split the problem
- Transpose the problem
- Apply data structures
- Find elementary algo



DE VINCI INNOVATION CENTER

Split the Problem

- Split the problem in smaller tasks
- Identify what each part should do
- Identify general blocks, their inputs, what they do
- Split the problem in multiple programs, sometimes written in different languages
    - In that case: How do they communicate ? File, Web, Api, UNIX Socket, IPC ?

Transpose the problem

- Use the right paradigm & tools for your problem:
  - For highly structured data, OOP can be a good idea
  - For simple problems, functional programming
  - For web application, other tools can be used
- For more complex problems, model them with math objects
- At this point, putting the general layout of your program on paper is a good idea

Apply Data Structures

- For each small function you have identified, see what data structure would work best
- Check the complexity of your functions

DE VINCI INNOVATION CENTER

Find Elementary Algorithms

- With the problem split down to elementary functions and the proper data structure, the algorithm should be much easier to find
- Don’t hesitate to split the problem again !