

# Projeto e Análise de Algoritmos

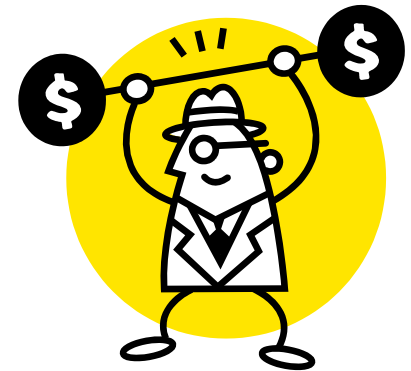
Prof. Marco Molinaro

[www.inf.puc-rio.br/~mmolinaro](http://www.inf.puc-rio.br/~mmolinaro)

[mmolinaro@inf.puc-rio.br](mailto:mmolinaro@inf.puc-rio.br)

# Boss assigns task:

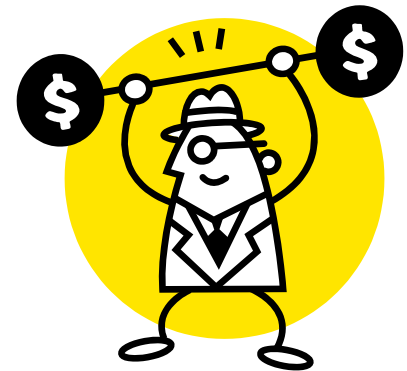
- Given the revenue per add; users profile; inventory per advertiser
- Which **ads** should the search engine exhibit to **maximize revenue**?



Everyday industry asks these questions

# Boss assigns task:

- Given bandwidth, processing capacity, number of available CPU's, memory, external storage
- Given constraints on how a crawler operates
- Make the most **efficient crawler**



Everyday industry asks these questions

# Possible answer

- Um? Tell me what to code.



**Bad:** Mundane programmers are not as valued as they use to be

# Your answer after **AA**

- I learned this great algorithm that will work.



**Outcome 1:** Know many different algorithms

# Your answer after **AA**

- I can develop a new algorithm for you



Distinguished professionals  
will always be needed

**Outcome 2:** Know **methods** for designing new algorithms

# Some questions

1. Write a regular expression which matches a email address
2. If you have 1 million integers, how would you sort them efficiently?
3. Given a file of 4 billion 32-bit integers, how to find one that appears at least twice?
4. Given an array, i) find the longest continuous increasing subsequence. ii) find the longest increasing subsequence
5. You are given with three sorted arrays (in ascending order), you are required to find a triplet (one element from each array) such that the distance is minimal
6. Given two linked lists, return the intersection of the two lists: i.e. return a list containing only the elements that occur in both of the input lists.
7. Describe the algorithm for a depth-first graph traversal

# Microsoft/Google questions

1. Write a regular expression which matches a email address
2. If you have 1 million integers, how would you sort them efficiently?
3. Given a file of 4 billion 32-bit integers, how to find one that appears at least twice?
4. Given an array, i) find the longest continuous increasing subsequence. ii) find the longest increasing subsequence
5. You are given with three sorted arrays (in ascending order), you are required to find a triplet (one element from each array) such that the distance is minimal
6. Given two linked lists, return the intersection of the two lists: i.e. return a list containing only the elements that occur in both of the input lists.
7. Describe the algorithm for a depth-first graph traversal



# Microsoft/Google questions

1. Write a regular expression which matches a email address
2. If you have 1 million integers, how would you sort them efficiently?
3. Given a file of 4 billion 32-bit integers, how to find one that appears at least twice?
4. Given an array, i) find the longest continuous increasing subsequence. ii) find the longest increasing subsequence
5. You are given with three sorted arrays (in ascending order), you are required to find a triplet (one element from each array) such that the distance is minimal
6. Given two linked lists, return the intersection of the two lists: i.e. return a list containing only the elements that occur in both of the input lists.
7. Describe the algorithm for a depth-first graph traversal

# Microsoft/Google questions

1. Write a regular expression which matches a email address
2. If you have 1 million integers, how would you sort them **efficiently**?
3. Given a file of 4 billion 32-bit integers, how to find one that appears at least twice?
4. Given an array, i) find the longest continuous increasing subsequence. ii) find the longest increasing subsequence
5. You are given with three sorted arrays (in ascending order), you are required to find a triplet (one element from each array) such that the distance is minimal
6. Given two linked lists, return the intersection of the two lists: i.e. return a list containing only the elements that occur in both of the input lists.
7. Describe the algorithm for a depth-first graph traversal

*Given 4 billion integers, find a number that appears at least twice (if exists)*

51	8	13	...	8	...	42	37
----	---	----	-----	---	-----	----	----

*Given 4 billion integers, find a number that appears at least twice (if exists)*

A= 

51	8	13	...	8	...	42	37
----	---	----	-----	---	-----	----	----

**Solution 1:** Try out **all pairs** of numbers (**brute force**)

```
For i=1 to A.len
```

```
  For j=i+1 to A.len
```

```
    If A[i] == A[j]
```

```
      Return A[i]
```

**Takes 80 milion sec.  
at 100 Gflops**

**Q:** How many pairs tested?

*Given 4 billion integers, find a number that appears at least twice (if exists)*

A= 

51	8	13	...	8	...	42	37
----	---	----	-----	---	-----	----	----

Can we do better and not compare all pairs?

**Solution 2:** For each number, search for its partner (bin. search)

```
sort(A)
For i=1 to A.len
  Binary search for
  If found
    Return A[i]
```

**Takes 1 sec. at  
100 Gflops**

**Q:** How many pairs tested?  $\approx$

We need to design **efficient** algorithms

**Brute force solution**

**Takes 80 milion sec.  
at 100 Gflops**

**Efficient solution**

**Takes 1 sec. at  
100 Gflops**

# Tentative Topics

# 1. Analysis of algorithms

## Basics

*How to formalize efficiency of an algorithm?*

## Sorting

*There is no other general sorting algorithm that makes fewer comparisons than Mergesort*

## Graphs

*Because of prerequisites, can I finish all remaining courses in 4 semesters?*



## 2. Design techniques

### **Greedy**

*With time conflicts, how should a university assign classrooms?*

### **Divide-and-conquer**

*More efficient multiplication of 2 numbers*

### **Dynamic programming**

*Compute similarity of 2 strands of DNA*

# 3. Additional topics

## Prediction with experts

*Given "experts" predicting stock movements, how to perform like the best expert?*

## Complexity

*What is P vs NP?*

# Useful Learning Techniques

# Read Ahead

- You are expected to read the lecture notes **before** the lecture.
- This will facilitate more productive discussion during class.



# Be Creative

- Ask questions
- Why is it done this way and not that way?



# Estrutura do curso

**Material:** slides de aula, listas de exercicios, (livro texto)

**Livro texto:** *Algorithm Design*, Kleinberg-Tardos

**Avaliacao:** 2 provas (+ prova final)

**Informações**

<http://www.inf.puc-rio.br/~mmolinaro>