

Projeto e Análise de Algoritmos

Prof. Marco Molinaro

www-di.inf.puc-rio.br/~mmolinaro

mmolinaro@inf.puc-rio.br

Your answer:

- I learned this great algorithm that will work.



Most known algorithms
are available in libraries

Your answer:

- I can develop a new algorithm for you.



Distinguished professionals
will always be needed

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 distance is minimum
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 distance is minimum
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

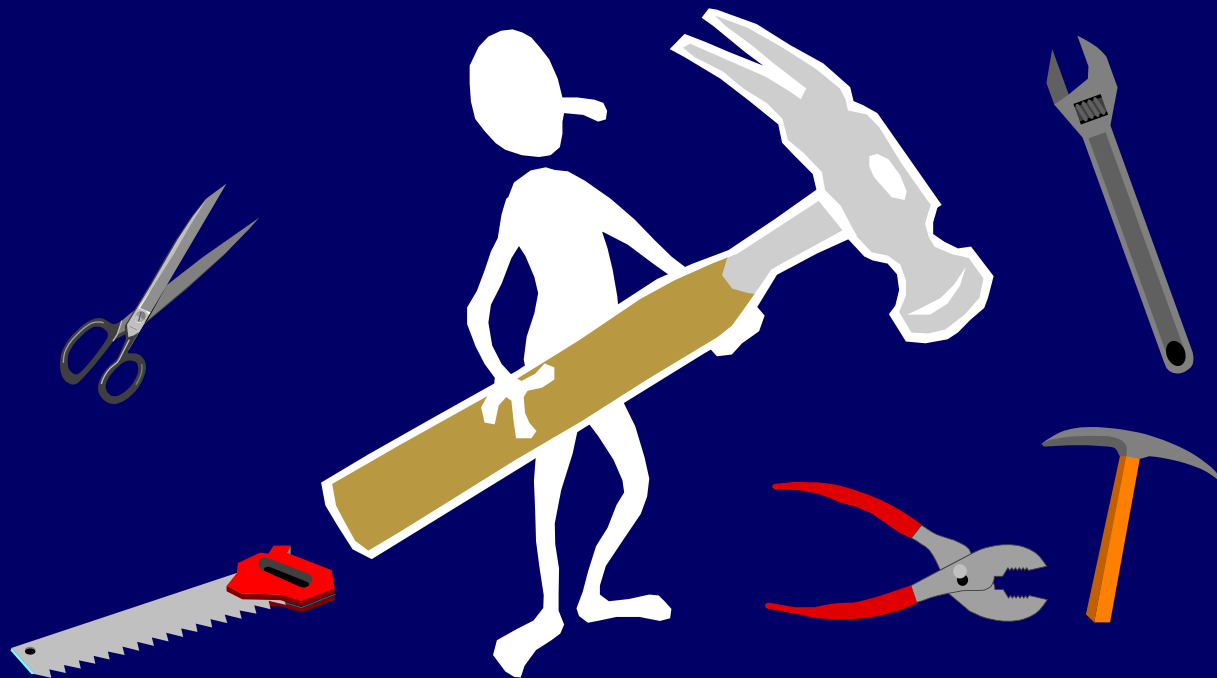
Course Content

- ~~A list of algorithms.~~
 - ~~– Learn their code.~~
 - ~~– Trace them until you are convinced that they work.~~
 - ~~– Implement them.~~

```
class InsertionSortAlgorithm extends SortAlgorithm
{
    void sort(int a[]) throws Exception {
        for (int i = 1; i < a.length; i++) {
            int j = i;
            int B = a[i];
            while ((j > 0) && (a[j-1] > B)) {
                a[j] = a[j-1];
                j--; }
            a[j] = B;
        }
    }
}
```

Course Content

- A survey of algorithmic design techniques.
- Abstract thinking.
- How to develop new algorithms for any problem that may arise.

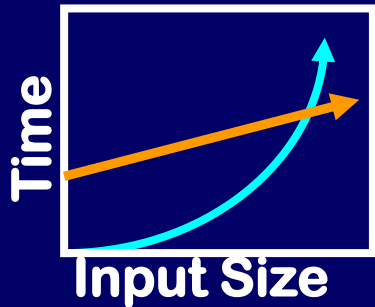


Tentative Topics

Start With Some Math

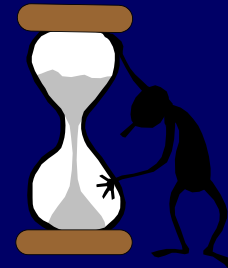
Classifying Functions

$$f(i) = n^{\Theta(n)}$$



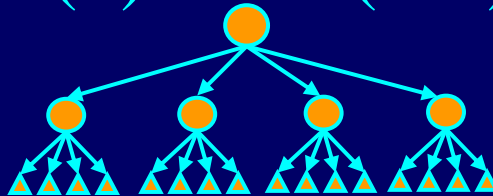
Time Complexity

$$t(n) = \Theta(n^2)$$



Recurrence Relations

$$T(n) = a T(n/b) + f(n)$$

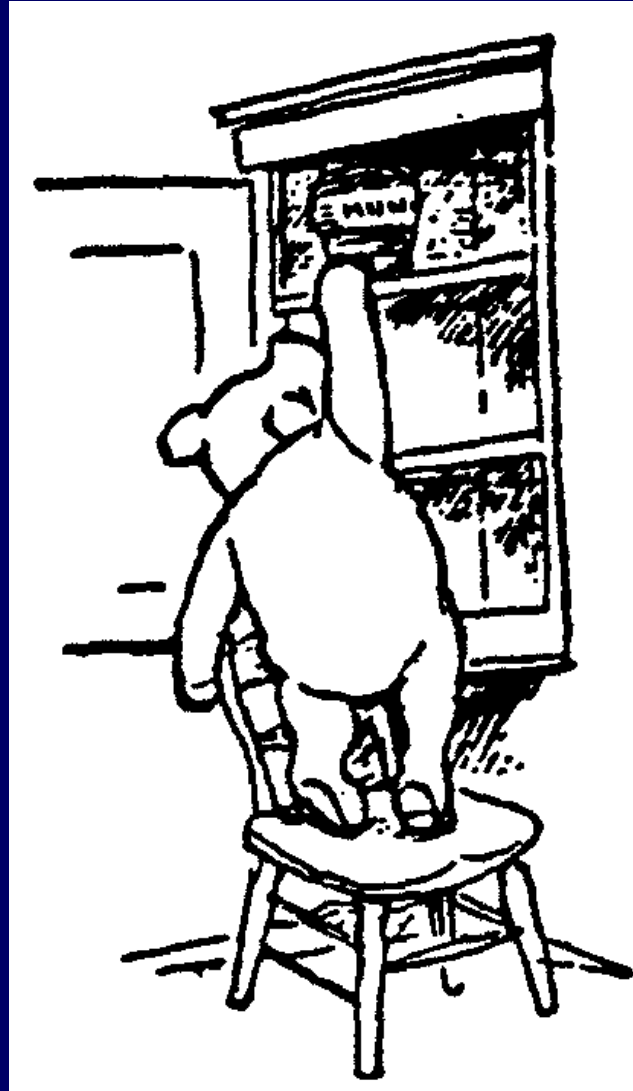


Use it in searching, sorting, merging

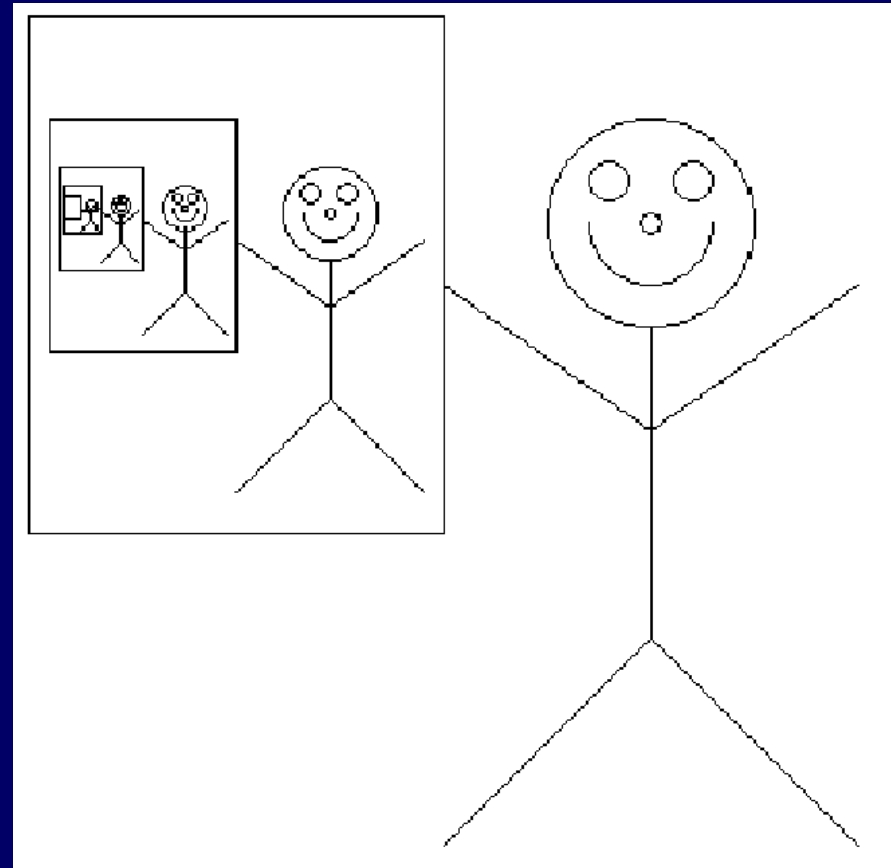
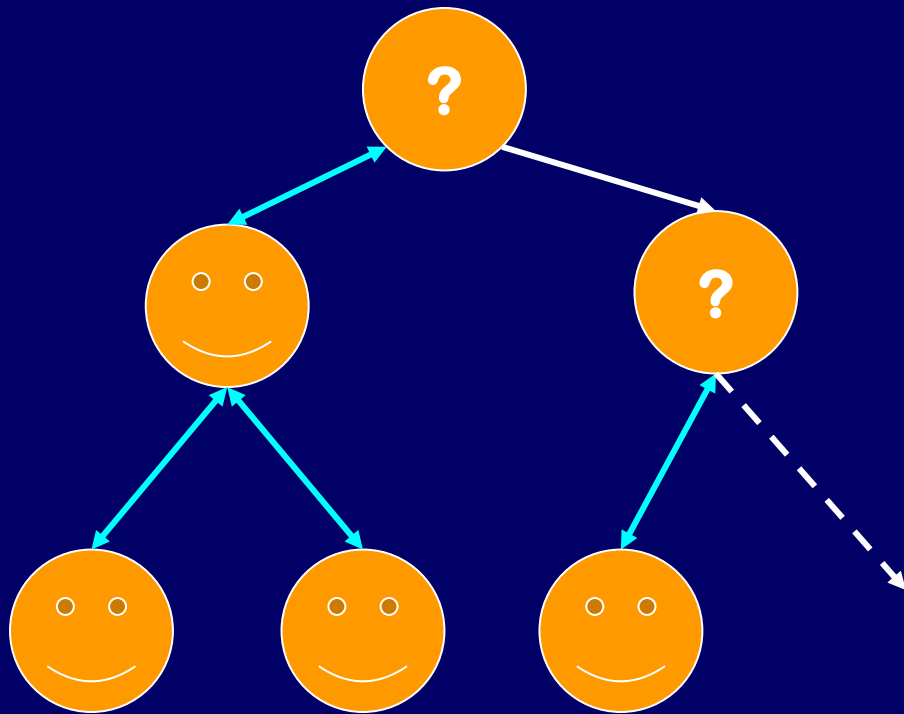
Graph Algorithms



Greedy Algorithms



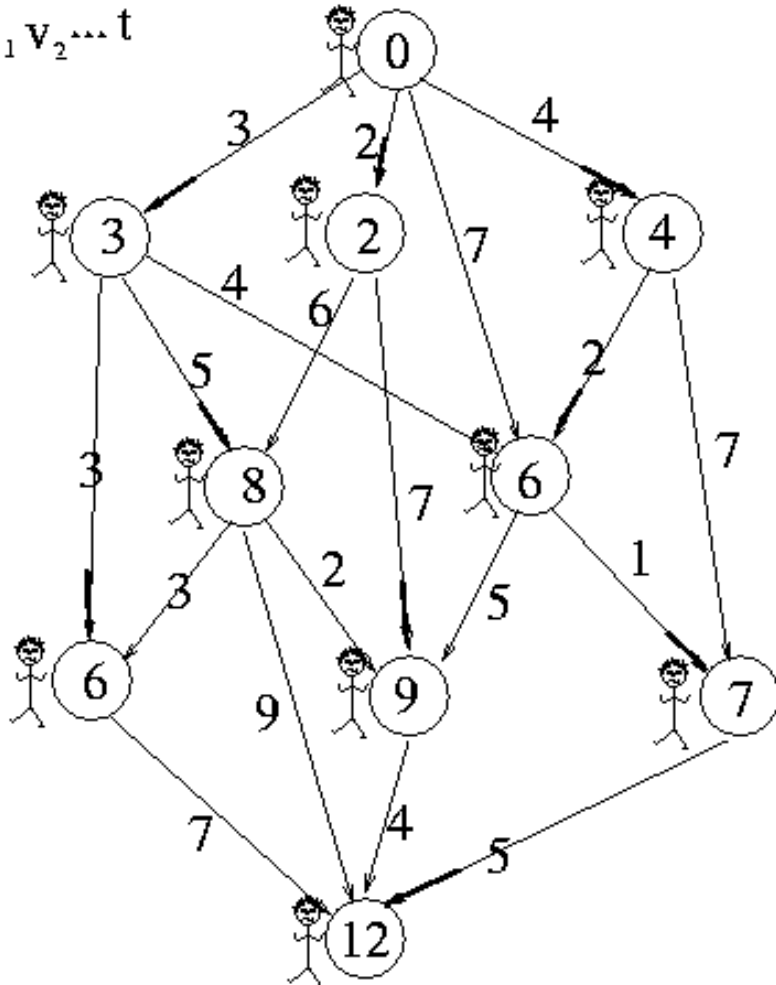
Divide and Conquer Algorithms



Dynamic Programming

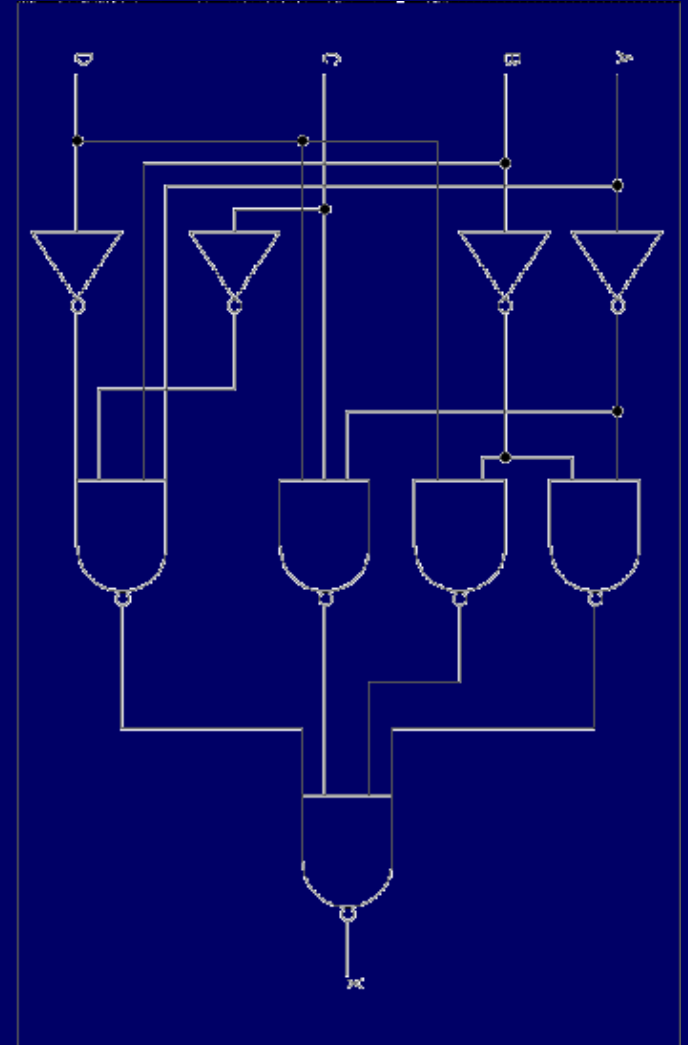
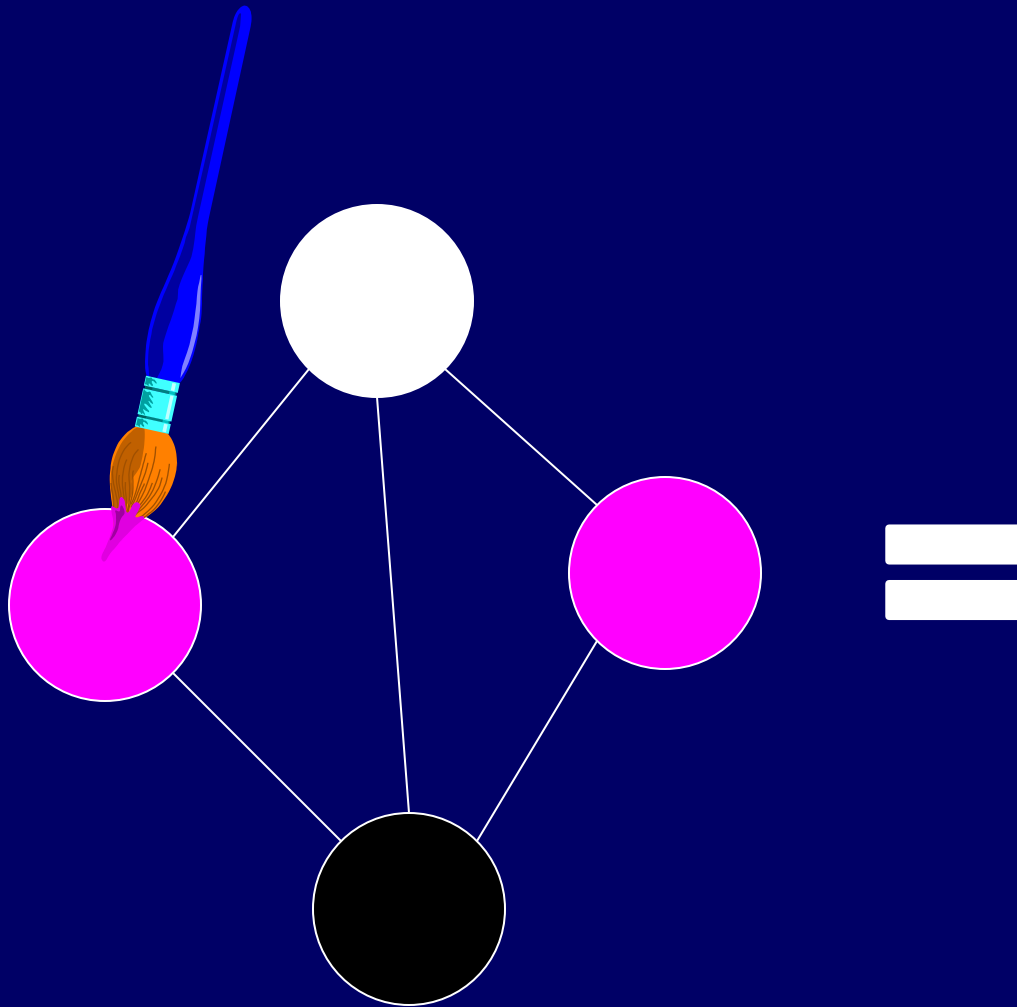
Solve each subinstance

$s \ v_1 \ v_2 \dots \ t$



	j	0	1	2	3	4	5	6	7	8
	y_j	0	1	0	1	1	0	1	0	
i	0	0	0	0	0	0	0	0	0	0
1	1	0	1	1	1	1	1	1	1	1
2	2	0	1	2	2	2	2	2	2	2
3	3	0	1	2	3	3	3	3	3	3
4	4	0	1	2	3	4	4	4	4	4
5	5	0	1	2	3	4	5	5	5	5
6	6	0	1	2	3	4	5	6	6	6
7	7	0	1	2	3	4	5	6	7	7

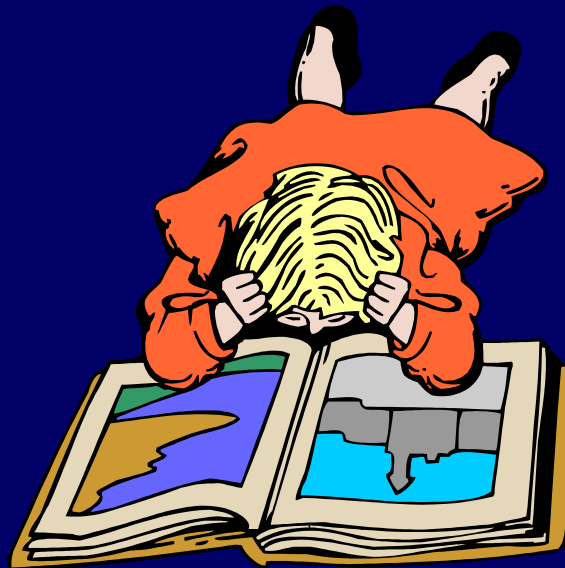
Complexity and Reductions



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?



Guesses and Counter-examples

- Guess at potential algorithms for solving a problem
- Look for input instances for which your algorithm gives the wrong answer

Refinement

The best solution comes from a process of repeatedly refining and inventing alternative solutions



Estrutura do curso

- Material: slides de aula, listas de exercicios, (livro texto)
- Livro texto: *Algorithm Design*, Kleinberg-Tardos
- 2 provas
- Informações

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