

2024-03-05: ancora buone caratterizzazioni (problema pirellone)

Si comincia a comporre il presente documento contestualmente e collaborativamente all'URL:

<https://etherpad.wikimedia.org/p/DiscreteOptimization2024-03-05>

Gli incontri avvengono sia in presenza che nella stanza Zoom:

<https://univr.zoom.us/j/83514538731>

E, quando disponibile, la loro registrazione è nel folder:

<https://univr.cloud.panopto.eu/Panopto/Pages/Sessions/List.aspx?folderID=c27d5938-629e-48c0-a238-b12a00b86b38>

Incontro 2024-03-05

Appunti da prima lezione del corso

WARNING: 3 people have the overlapping problem with the class of Albi

How many come from Mathematics: 6

How many come from Data Science: 17

How many come from Artificial Intelligence: 4

Friday: mostly done by Roberto online.

(We put all the links in the Telegram Group)

Tuesday: mostly done by Romeo in classroom (but anyhow videorecorded)

EXAM:

last edition it was a project + an oral exam

for this edition, my goal is to:

- homeworks (we can not make these compulsory but we strongly recommend trying to follow up the course while ongoing):

to be done during the course will already begin collecting points for you. (TALight infrastructure)

- project will be done on the same infrastructure (which provides immediate feedback)

- written exam

=====

DECISION MAKING and DISCRETE OPTIMIZATION

The aim is to design systems that help us in taking decisions and in optimizing any possible system. We want the best.

Are issues in the realm of Discrete Mathematics (which is that subset of Mathematics that we call Informatics).

A decision problem is a problem where:

INPUT: any string/number

OUTPUT: either Yes or No (0 or 1)

Decision problems are also called Recognition problems.

Def: a language is any subset of the set of all possible strings.

Fact: The union of an infinite countable number of countable sets is countable.

Therefore, the rational numbers are countable.

Doubt: maybe even the real numbers are countable?

But then I found out a way to prove they are not! (diagonalization)

Consider the following square 0,1 matrix:

	b1	b2	b3	b4
x1	1	1	0	1
x2	0	1	0	1
x3	1	0	1	1
x4	1	0	0	0

Where does the diagonal intersect row i?

Where does the diagonal intersect row 3?

The 3-th element of the diagonal is precisely the 3-th element of the 3-th row.

The i-th element of the diagonal is precisely the i-th element of the i-th row.

Therefore, the complement of the diagonal differs from the i-th row on their i-th element.

	1	2	3	4
S1	1	1	0	1
S2	0	1	0	1
S3	1	0	1	1
S4	1	0	0	0

Does it have any row which is identical to the inverse of the main diagonal?

Diagonal: 1 1 1 0 The inverse of the diagonal: 0 0 0 1

How many are the possible decision problems?

Precisely as many as the languages.

Precisely as many as the subsets of the natural numbers.

Ah, they are not countable!

proof: based on the diagonalization technique of Cantor.

Therefore: languages and decision problems are essentially the same things (the recognition of a language is a decision problem).

Turing - Church:

- there exists a universal machine:

- consider your PC, you can give it any program, and it will run it on any

possible input string.

- as a consequence of this (via the diagonalization technique of Kantor) we discover that not only there are decision problems that have no algorithm (non decidable) but we can actually define a specific problem that has no algorithm.

INPUT: set S of 101 numbers (possible with repetitions)

TASK: find one s in S such that s \geq at least 50 numbers in S
AND s \leq at least 50 numbers in S

Enough for a feeling on decidability theory

We will speak more widely about complexity theory in future classes.

PROBLEMS=MODELS

The sum of the ages of my children is 7

Their difference is 1

What are the ages of my children?

Could it be 3 and 4?

This should be easy to verify:

3+4=7 ok

4-3=1 ok

so, that's it!

Problems that when we find a solution that solution is compact and can be verified (and put to practice) in reasonable time!

P decision problems that can be solved in polynomial time

NP decision problems that can, when the answer is YES, a compact certificate can be given that can be checked in polynomial time

coNP decision problems that can, when the answer is NO, a compact certificate for the NO can be given that can be checked in polynomial time

Maybe tomorrow:

The sum of the ages of my children is 11

Their difference is 3

What are the ages of my children?

ok, then I consider the following "template" of a problem:

Maybe tomorrow:

The sum of the ages of my children is S

Their difference is D

What are the ages of my children?

This template for a question is what we consider to be an algorithmic problem.

These problem templates are also called MODELS.

$|A| = \sum (\text{sign}(\pi_i) * \prod (A[i, \pi_i(i)] : i=1,2,\dots,n) : \text{all permutations } \pi_i \text{ of } (1,2,\dots, n))$

$\text{sign}(\pi_i) = 1$ if π_i has an even number of inversions

$\text{sign}(\pi_i) = -1$ if π_i has an odd number of inversions