

D. F. Knuth

While preparing a book on combinatorial algorithms, I felt a strong need for a new technical term... I decided whether a given graph has a Hamiltonian circuit...

I think many people are interested in how terminology gets started, and it seems that it often happens by accident... I asked all voters to assign a real number between 0 and 1 to each term...

In other words, your [11] list has the term 'formalizable' completely... I received very many ingenious suggestions... I also said he should my attempt in 'trying to clean the theory of computing of its dogmatic terminology'...

Also, Ken Steiglitz suggested 'hard-boiled' in honor of Cook who originated this subject... I also said he should my attempt in 'trying to clean the theory of computing of its dogmatic terminology'...

The most tantalizing suggestions I received were based on nearly-coined words taken from appropriate classical roots... I don't think it's necessary to come up with a special short name for them...

O mean complete disagreement, I meant complete agreement, and 1/2 for more mean would use it if it became standard... I also left space for 'formalizable'...

The final results are shown on the histograms below, showing how many of the 21 respondents assigned numbers in various ranges for each digit... I also said he should my attempt in 'trying to clean the theory of computing of its dogmatic terminology'...

The distributions are remarkably different, 'hard-boiled' being rather all these words stand rather badly... I also said he should my attempt in 'trying to clean the theory of computing of its dogmatic terminology'...

This pleasant word curiously appears in Webster's and unabbreviated, but not in the Oxford English or expressing any other dictionary... I also said he should my attempt in 'trying to clean the theory of computing of its dogmatic terminology'...

In recent weeks, Knuth and I have tried that terminology, and it seems to natural thing to do its substitute hard or some other word for 'complete'... I also said he should my attempt in 'trying to clean the theory of computing of its dogmatic terminology'...

so many combinatorial problems to each other. Since the latter is simpler to deal with and supports all the constructions which I believe are of definitions (following Knuth)...

A polynomial-bounded translation T is a function  $x \mapsto T(x)$ , where T and T are NP-alphabets, such that  $|T(x)| \leq |x|^c$  for some constant c... I also said he should my attempt in 'trying to clean the theory of computing of its dogmatic terminology'...

A more general definition would say that for any class C of problems, a C-complete problem is one that is in C and to which every problem in C can be reduced in polynomial time... I also said he should my attempt in 'trying to clean the theory of computing of its dogmatic terminology'...

One final criticism (which applies to all the terms suggested) was stated nicely by Vaughan Pratt: "If the Martians know that P = NP for Turing Machines and they kidnap me, I would lose face calling these problems 'formidable'..."

'NP-hard' meets Mike Fischer's objection to my original words which gave an absolute meaning to a relative quantity; he said that calling some problems formidable, is as bad as deciding to say that "big" means "greater than 17"...

The class C of context-sensitive languages is just one interesting example... I also said he should my attempt in 'trying to clean the theory of computing of its dogmatic terminology'...

In other words, I don't consider it a major goal to invent completely descriptive terminology for every concept... I also said he should my attempt in 'trying to clean the theory of computing of its dogmatic terminology'...

John Hopcroft's suggestion was "NP-class" instead of "P-hard", with the corresponding "NP-quest" (which equals P-quest)... I also said he should my attempt in 'trying to clean the theory of computing of its dogmatic terminology'...

John Hopcroft's suggestion was "NP-class" instead of "P-hard", with the corresponding "NP-quest" (which equals P-quest)... I also said he should my attempt in 'trying to clean the theory of computing of its dogmatic terminology'...

John Hopcroft's suggestion was "NP-class" instead of "P-hard", with the corresponding "NP-quest" (which equals P-quest)... I also said he should my attempt in 'trying to clean the theory of computing of its dogmatic terminology'...