

Re: Is there a polynomial algorithm for that problem?

Re: Is there a polynomial algorithm for that problem?

Source: <http://sci.tech-archive.net/Archive/sci.math/2008-07/msg02665.html>

- *From:* fjblurt@xxxxxxxxxx
 - *Date:* Sat, 26 Jul 2008 03:07:04 -0700 (PDT)
-

On Jul 25, 5:07 am, wolfgang.woege...@xxxxxxxxxx wrote:

Hi folks,

I want to solve the following problem:

There is a set of n numbers that can be either positive, negative or zero. Now I need to split that numbers into groups of 14 elements. If n is a multiple of 14 then all groups have 14 elements. Otherwise, all but one groups have 14 elements

The question that is to be answered is the following: Is there a way to group the numbers into groups in a way that no sum of the numbers of a group is negative? And is there is a solution, how do the groups look like (this is called the solution in the following)?

There are two very easy cases:

1. The sum over all n numbers is negative. Then it is clear that there will be no way to group the numbers.
2. Every number is positive. Then, no matter what grouping is chosen, that grouping is a solution.

But, if the sum over all n numbers is positive, but this is no guarantee that there is a solution.

An algorithmic easy way to find out if there is solution is the following: Construct every possible set and check if the sum of each set is non-negative. Unfortunately that algorithm does not scale. While there are only 4.5 million possible sets for $n=25$, there are already 10^{81} possible sets for $n=100$.

There are many heuristics that return in most cases a solution if there exists one. But what I am looking for is an algorithm that scales polynomially in the worst case and ALWAYS returns a solution if there exists one.

Can anyone tell me whether there is such an algorithm or argue why such an algorithm cannot exist?

Re: Is there a polynomial algorithm for that problem?

Re: Is there a polynomial algorithm for that problem?

Thanks in advance,
Wolfgang

Your problem is NP-complete, so it is very likely that no such algorithm exists.

Here is a reduction from 3-Partition. See http://en.wikipedia.org/wiki/3-partition_problem

An instance of 3-Partition consists of a list of $3m$ positive integers summing to mB , where B is some positive integer, and asks whether the list can be partitioned into m lists of size 3, each of which sums to B . Without loss of generality, assume that B is not divisible by 11. (If it is, then add 1 to each of the integers to get an equivalent instance which has $B' = B+3$.) We can transform this into an instance of your problem by taking this list and adding $11m$ copies of the number $-B/11$, which is not an integer. Then if your problem has a solution, the new list must partition into lists of size 14, all of which sum to zero. Each of them must have exactly 11 copies of $-B/11$, since the other numbers are positive integers, so the remaining three sum to B .

Therefore, if it were possible, in polynomial time, to even determine whether an arbitrary instance of your problem has a solution, it would also be possible to determine, again in polynomial time, whether an arbitrary instance of 3-Partition had one. 3-Partition is known to be NP-complete, so that would imply that every NP problem also has a polynomial time algorithm, i.e. $P=NP$, which is strongly believed to be false.

Neat problem! Thanks for posting.