Класс NP.

20 Февраля 2018

1. У Вас есть несколько чисел, битовая длина каждого не первосходит k. Затем вы запускаете алгоритм который выполняет n арифметических операции(умножение, сложение, деление, вычитание) какой максимальной битовой длины может получиться конечный ответ?

2. Предположим $L_1, L_2 \in NP$, что вы можете сказать про языки $L_1 \cup L_2, L_1 \cap L_2$ относительно их принадлежности NP.

3. Покажите, что язык PRIMES, состоящий из бинарных строек задающих простые числа, принадлежит классу NP. Заметим, что p – простое ТиТТ когда для любого простого делителя числа $p-1$ существует $a \in \{2, \dots p-1\}$ удовлетворяющее тому, что $a^{p-1} = 1 \bmod p$, но $a^{\frac{p-1}{q}} \neq 1 \bmod p$.

4. Существует ли алгоритм, проверяющий, работает ли данная программа(машина Тьюринга) полиномиальное время? (т.е. проверяет, что существует какой-то полином p, что программа работает на любом входе x не больше $p(|x|)$).

   Определение: co-NP — класс языков, которые являются дополнениями языков из **NP**. То есть если $L \in NP$, то $\{0,1\}^* \setminus L \in$ **co-NP**. Другими словами вместо подсказки(сертификата) о принадлежности слова языку у нас должна быть подсказка(сертификат) о непринадлежности слова языку. Множество выполнимых булевых формул является языком из NP, поскольку есть сертификат в виде выполняющего набора. В свою очередь, множество булевых тавтологий(всегда истинных формул) принадлежит классу **co-NP**, поскольку есть сертификат в виде невыполняющего набора, показывает непринадлежность слова языку.

5. Покажите, что $P \subseteq NP \cap coNP$.

6. Покажите, что если $P = NP$, то $NP = coNP$.

7. Покажите, что если $P = NP$, то вы можете найти выполняющий набор за полиномиальное время. Заметим, что сам факт $P = NP$, гарантирует только, что вы можете определить сущетвует ли выполняющий набор но не найти его.

8. Покажите, что если $P = NP$, то вы можете разложить число на простые множители за полиномиальное время.