Last name	Name	ID

Generic Competence EDA

- *This exercise must be handed in on 09/06/2017 at the* **beginning** *of the final exam.*
- *Write your full name and ID.*
- *Write your answers within the reserved space.*
- *When giving a reference to a source (book, journal, website, etc.), follow the style "ISO-690 (author-date, English)". You can generate references in this format at:* `www.citethisforme.com/guides/iso690-author-date-en`
- *When giving an URL, please write* ***clearly*** *and use Google URL Shortener:* `https://goo.gl`

(a) Define the complexity class co-NP.

(b) Give the reference of a source where this class is defined.

(c) Show that P $\subseteq$ co-NP.

(d) From the previous exercise and that P $\subseteq$ NP one can deduce that P $\subseteq$ NP $\cap$ co-NP. An important open problem in theoretical Computer Science is whether P = NP $\cap$ co-NP or not.

For a long time, a problem from NP $\cap$ co-NP that was not known to belong or not to P (and hence was candidate to refute that P = NP $\cap$ co-NP) was the problem of **PRIMES**: given a natural number (represented by the vector of its digits in binary), to determine whether it is prime or not. Finally however it was shown that this problem belongs to P.

Give the reference of a source where it is **proved** that **PRIMES** belongs to P.

(e) Another important open problem in theoretical Computer Science is whether NP = co-NP. If NP $\neq$ co-NP, what can we say then about the problem P = NP? Justify your answer.