

⑦ Factors: Donada $N \in \mathbb{Z}^+$ $\rightarrow X_i \in \text{Primes}$
 retornar $x_1^{a_1}, \dots, x_n^{a_n}$ tq $x_1^{a_1} * \dots * x_n^{a_n} = N$

Factorization: Donada $N \in \mathbb{Z}^+$
 retornar p tq $1 < p < N \wedge N \bmod p = 0$

Factorization-D: Donades $N \in \mathbb{Z}^+$ i $D < N$
 retornar $\exists p$ tq $1 < p < D \wedge N \bmod p = 0$?

a) Factorization-D is NP and co-NP

En ambdós casos podem donar un algorisme verificador que donada una possible solució (testimoni) retorni si és correcte o no:

NP:

input N, D, p
 si $N \% p = 0$ and $p < D$
 return TRUE
 else
 return FALSE

}
 La correctesa és trivial

co-NP:

El problema complementari ha de pertànyer a NP:
 $\overline{\text{Fact-D}}$: Donats N i D
 retornar si $\forall p \leq D, N \% p \neq 0$
 ho podem comprovar amb el següent verificador:

input $x_1, \dots, x_n, a_1, \dots, a_n, D, N$
 for $i = 1$ to n
 if $N \% x_i = 0$ and $x_i < D$
 return false
 if $N \% x_i \neq 0$
 return false
 }
 return TRUE
 }

El verificador retornarà fals si troba algun divisor de N més petit que D .

b) Si $P=NP$, demostrar que Factorization $\in FP$

Com hem demostrat en l'apartat a) que Fact-D és NP, i $P=NP$, existeix un algoritme que decideix Fact-D en temps polinòmic, per tant, es pot fer una cerca dicotòmica per trobar un divisor de N :

```

input N {
  D = N/2
  while Fact-D(N, D) {
    D = D/2
  }
  for i = 2 to D {
    if N % i == 0 return i
  }
  return PRIME
}

```

$\in P=NP$

Aquest algoritme fa una cerca dicotòmica per a trobar el primer més petit que divideix N , el cost, al fer cerca binària és: $O(\log_2(N)) \rightarrow O(\log_2(2^{|N|})) \rightarrow O(|N|)$.

El bucle del final fa una quantitat logarítmica de passos respecte a N , ja que es fan un nombre primer de passos. Relativament

c) Si $P=NP$, Factors $\in FP$:

Similar a l'apartat anterior, si $P=NP$, tenim un algoritme que resol Factorization en $\in P$, i podem construir:

```

input N {
  Lps = []
  while N != 1 {
    np = Factorization(N)
    Lps = Lps U np
    N = N / np
  }
  return Lps
}

```

$\in P=NP$

potar donar error (np=PRIME)

Aquest algoritme extreu una llista de tots els factors primers de N , fa un nombre relativament primer d'iteracions, per tant és logarítmic respecte N : $O(\log_2(N)) \rightarrow O(\log_2(2^{|N|})) \rightarrow O(|N|)$

Si N és primer, l'algoritme donarà error en la línia marcada.