Igor Kenzo Miyamoto Dias

ADO - 04

2) Mostre que P é fechado em concatenação

Suponha que tenha uma linguagem $L_1 \in P$ e outra $L_2 \in P$, então existe uma MT de tempo polinomial M_1 que decide L_1 e uma M_2 que decide L_2. Por se tratar de MT de tempo polinomial, podemos supor que as máquinas tem tempo $O(n^{k1})$e $O(n^{k2})$ para M_1 e M_2, respectivamente, onde n é o tamanha da entrada e k1 e k2 são constantes.

Podemos criar uma M3 que decide a concatenação:

M3 = " Sobre a entrada $w = c1, c2, ..., cn$, onde cada c_i é um caracter de w e $c_i \in \Sigma$.

1. Para cada i = 0,1,2,..., n, faça:
2. Rode sobre M_1 a entrada $w_1 = a_1 a_2 ... a_i$ e

   rode sobre M_2 a entrada $w_2 = a_{i+1} a_{i+2} ... a_n$,

   Se ambos aceitarem, Aceite.
3. Se em nenhuma das iterações aceitar. Rejeite."

M3 testa todas as combinações que w pode estar concatenado, separando em w1 e w2 e checando se estão em L(M1) e L(M2), respectivamente.

O looping que divide a entrada roda no máximo n + 1 vezes, então O(N).

Como |w1| ≤ |w| e |w2| ≤ |w|, são $O(n^{k1})$e $O(n^{k2})$, logo a complexidade dessa parte é $O(n^{max(k1,k2)})$.

Então o tempo final é $O(n) * O(n^{max(k1,k2)})$ = $O(n^{1+max(k1,k2)})$, o que é polinomial.

c.q.d

3) Mostre que NP é fechado em concatenação

Suponha que tenha uma linguagem $L_1 \in NP$ e outra $L_2 \in NP$, então existe uma MT não determinística de tempo polinomial M_1 que decide L_1 e uma M_2 que decide L_2.

Podemos criar uma M3 que decide a concatenação:

M3 = " Sobre a entrada $w = c1, c2, ..., cn$, onde cada c_i é um caracter de w e $c_i \in \Sigma$.

1. De forma não determinística, corte a entrada w de todas as maneiras possíveis em w1 e w2.
2. Rode sobre M_1 a entrada w_1 e

   rode sobre M_2 a entrada w_2,

   Se ambos aceitarem, Aceite.
3. Se em nenhuma das iterações aceitar. Rejeite."

M3 testa todas as combinações que w pode estar concatenado não deterministicamente, separando em w1 e w2 e checando se estão em L(M1) e L(M2), respectivamente.

O looping que divide a entrada roda no máximo n vezes, então O(n).

Como |w1| ≤ |w| e |w2| ≤ |w|, são $O(n)$ por rodar não deterministicamente, logo a complexidade dessa parte é $O(n)$, o que é tempo polinomial.

c.q.d