

UFMS – DEPARTAMENTO DE COMPUTAÇÃO E ESTATÍSTICA
BACHARELADO EM CIÊNCIA DA COMPUTAÇÃO

Coplexidade Computacional

Prof. Marco Aurélio

LISTA 2 – ENTREGA: ATÉ 01/10 ÀS 19H NA SECRETARIA DO DCT

1. Descreva uma máquina de Turing para calcular:
 - (a) $\lceil \log n \rceil$
 - (b) $n + m$, tal que a entrada é codificada da forma $\triangleright x + y$, onde x e y são as representações binárias de n e m e o $+$ é um símbolo especial do alfabeto.
2. Considere a linguagem $L = \{M \mid M \text{ sem entrada escreve três 1's consecutivos em alguma fita}\}$. L é recursiva? L é recursivamente enumerável?
3. Suponha que máquinas de Turing possam *remove* e *insere* símbolos nas fitas, ao invés de somente escrever.
 - (a) Descreva cuidadosamente a função de transição e a computação de tal máquina.
 - (b) Mostre que se tal máquina opera em tempo $f(n) \geq n + 2$, então ela pode ser simulada por uma máquina de Turing convencional em tempo $O(f^2(n))$.
4. Prove que se L é uma linguagem em **SPACE**($f(n)$), então para qualquer $\epsilon > 0$, $L \in \mathbf{SPACE}(2 + \epsilon f(n))$.
5. Mostre que se L é recursivamente enumerável então existe uma máquina de Turing que enumera L sem nunca repetir um elemento de L .