TEC0001 – Teoria da Computação
Aula 10
Complexidade de Tempo

Karina Girardi Roggia

karina.roggia@udesc.br

Departamento de Ciência da Computação
Centro de Ciências Tecnológicas
Universidade do Estado de Santa Catarina

2020

Máquina 1

M_1 = Sobre a entrada w:

1. Faça uma varredura na fita e *rejeite* se for encontrado algum 0 à direita de algum 1.
2. Repita se existem ambos, 0s e 1s, na fita:
3. Faça uma varredura na fita, cortando um único 0 e um único 1.
4. Se ainda permanecerem 0s após todos os 1s tiverem sido cortados ou se ainda permanecerem 1s após todos os 0s tiverem sido cortados, *rejeite*. Caso contrário, se não houver 0s nem 1s sobre a fita, *aceite*.

Máquina 2

$M_2 =$ Sobre a entrada w:

1. Faça uma varredura na fita e *rejeite* se for encontrado algum 0 à direita de algum 1.
2. Repita se existem ambos, 0s e 1s, na fita:
3. Faça uma varredura na fita, verificando se o número total de 0s e 1s remanescentes é par ou ímpar. Se for ímpar, *rejeite*.
4. Faça uma varredura novamente na fita, cortando alternadamente um 0 sim e outro não começando com o primeiro 0, e, então cortando alternadamente um 1 sim e outro não começando com o primeiro 1.
5. Se nenhum 0 e nenhum 1 permanecerem na fita, *aceite*. Caso contrário, *rejeite*.

M_3 = Sobre a entrada w:

1. Faça uma varredura na fita e *rejeite* se for encontrado algum 0 à direita de algum 1.
2. Faça uma varredura nos 0s sobre a fita 1 até o primeiro 1. Ao mesmo tempo, copie os 0s para a fita 2.
3. Faça uma varredura nos 1s sobre a fita 1 até o final da entrada. Para cada 1 lido sobre a fita 1, corte um 0 sobre a fita 2. Se todos os 0s estiverem cortados anter que todos os 1s sejam lidos, *rejeite*.
4. Se todos os 0s tiverem sido cortados, *aceite*. Se restar algum 0, *rejeite*.

Definição (Tempo de Execução)

Seja M uma máquina de Turing determinística que para sobre todas as entradas. O **tempo de execução** ou **complexidade de tempo** de M é a função $f : \mathbb{N} \to \mathbb{N}$, onde $f(n)$ é o número máximo de passos que M usa sobre entradas de comprimento n.

Se $f(n)$ for o tempo de execução de M, dizemos que:

- M roda em tempo $f(n)$
- M é uma máquina de Turing de tempo $f(n)$

Definição (Notação $\mathcal{O}$)

Sejam f e g funções $f, g : \mathbb{N} \rightarrow \mathbb{R}^+$. Dizemos que $\mathbf{f(n) = \mathcal{O}(g(n))}$ se existem inteiros positivos c e n_0 tais que para todo inteiro $n \geq n_0$

$$f(n) \leq cg(n)$$

Quando $f(n) = \mathcal{O}(g(n))$ dizemos que $g(n)$ é um **limitante superior assintótico** para $f(n)$.

Intuitivamente: f é menor ou igual a g se desconsideradas as diferenças até um fator constante.

"Equações $\mathcal{O}$"

Apesar de se usar o símbolo $=$, em expressões com a notação $\mathcal{O}$ a leitura deve ser feita **somente da esquerda para a direita**.

$f_1(n) = 5n^3 + 2n^2 + 22n + 6$

$\quad f_1(n) = \mathcal{O}(n^3)$

$\quad f_1(n) = \mathcal{O}(n^4)$

$\quad f_1(n)$ **não é** $\mathcal{O}(n^2)$

$f_2(n) = 3n \log_2 n + 5n \log_2 \log_2 n + 2$

$\quad f_2(n) = \mathcal{O}(n \log n)$

$f(n) = \mathcal{O}(n^2) + \mathcal{O}(n) \Rightarrow f(n) = \mathcal{O}(n^2)$

Definição (Classe de Complexidade de Tempo)

Seja $t : \mathbb{N} \to \mathbb{R}^+$ uma função. A classe de complexidade de tempo $\mathbf{TIME(t(n))}$ é a coleção de todas as linguagens que são decidíveis por uma máquina de Turing de tempo $\mathcal{O}(t(n))$.

Máquina de Turing Não-Determinística Decisora

Definição (Tempo de Execução de MTND Decisora)

Seja N uma máquina de Turing não-determinística decisora. O **tempo de execução** de N é a função $f : \mathbb{N} \to \mathbb{N}$ onde $f(n)$ é o número máximo de passos que N usa sobre qualquer ramos de sua computação sobre qualquer entrada de comprimento n.

Máquina Determinística versus Máquina Multifita

Teorema

Seja $t(n)$ uma função onde $t(n) \geq n$. Então todas máquina de Turing multifita de tempo $t(n)$ tem uma máquina de Turing determinística de uma única fita equivalente de tempo $\mathcal{O}(t^2(n))$.

Ideia geral: simulação da multifita em uma única fita executará $t(n)$ passos da multifita onde cada passo na simulação terá um tempo $\mathcal{O}(t(n))$.

Complexidade entre Modelos de MTs

Máquina Determinística versus Máquina Não Determinística

Teorema

Seja $t(n)$ uma função onde $t(n) \geq n$. Então para toda máquina de Turing não-determinística de tempo $t(n)$ existe uma máquina de Turing determinística de uma única fita equivalente de tempo $2^{\mathcal{O}(t(n))}$.

Ideia geral: simulação da não-determinística em uma única fita executará a visita em $\mathcal{O}(b^{t(n)})$ folhas da árvore de computação onde cada passo na simulação terá um tempo $\mathcal{O}(t(n))$. (b é o número máximo de escolhas na função programa da máquina não-determinística)

Como $b^{t(n)} = 2^{\log_2 b^{t(n)}} = 2^{t(n)\frac{\log b}{\log 2}}$:

$$\mathcal{O}(t(n)b^{t(n)}) = \mathcal{O}(t(n)2^{\mathcal{O}(t(n))}) = \mathcal{O}(2^{\log_2 t(n)}2^{\mathcal{O}(t(n))}) = 2^{\mathcal{O}(t(n))}$$