

Trabalho 1 - Teoria da Computação

Gabriel Leal C. Amaral, Gustavo F. Olegário

October 2016

1 Máquina 1a - Fita única

1.1 Enunciado da linguagem

$$L = \{w\#w \mid w \in \{a,b,c\}^*\}$$

1.2 Descrição do algoritmo

1. No estado inicial, se ler a entrada vazia, aceite a palavra. Caso contrário, vá para o próximo estado.
2. Marcar a posição lida com x caso seja a ou b.
3. Percorrer até o primeiro elemento após a cerquilha que seja diferente de x.
4. Caso esse elemento seja diferente do último marcado com x no lado esquerdo da cerquilha, rejeite. Caso contrário, marque com x e volte até o último elemento marcado no lado esquerdo da cerquilha. Se o elemento seguinte for x, aceite a palavra. Caso seja a ou b, vá para 2. Caso contrário, rejeite.

As imagens dos testes estão em /Imagens/1a A codificação da máquina está em machines.jflap

2 Máquina 2a - Multifita

2.1 Enunciado da linguagem

$$L = \{ww \mid w \in \{0,1\}^*\}$$

2.2 Descrição do algoritmo

1. No estado inicial, existem três possibilidades. Se for uma palavra do alfabeto inicial, 0 ou 1, marca como lida com X ou Y, respectivamente. Caso contrário, segue a sequência de estados para o estado final

2. Se no estado inicial, a máquina tiver lido 1 ou 0, a máquina prosseguirá até encontrar uma palavra vazia.
3. Ao encontrar uma palavra vazia, irá retroceder até encontrar 1 ou 0.
4. Se encontrar um 1 ou 0, marcará como lido e irá retroceder até encontrar uma palavra não lida. Quando achar que for lida, avança uma casa a frente.
5. Caso tenha seguido o caminho para o estado final, para cada elemento já lido na fita 1, marcará com um símbolo diferente e escreverá o antigo símbolo na fita 2.
6. Por último, percorrerá a ambas as fitas até a extremidade esquerda, garantindo que as duas fitas tem a mesma palavra. As imagens dos testes estão em /Imagens/2a

3 Máquina 2b - Multifita

3.1 Enunciado da linguagem

$$L = \{ ww^R w \mid w \in \{a,b\}^* \}$$

3.2 Descrição do algoritmo

1. No estado inicial, se ler a entrada vazia, aceite a palavra. Caso contrário, vá para o próximo estado.
2. Copie cada entrada lida da fita 1 para a fita 2. Ao chegar na entrada vazia, volte uma posição à esquerda em ambos os cabeçotes.
3. Mova o cabeçote da fita 1 para esquerda. Mova o cabeçote da fita 2, 3 vezes para a esquerda.
4. Repita o passo anterior até o cabeçote 2 chegar à entrada vazia. Chegando na entrada vazia, mova o cabeçote das fitas 1 e 2 para a direita. Neste ponto, o cabeçote 1 aponta para o início da última string e o cabeçote dois para o começo da primeira.
5. Percorrer até o final da string analisada na fita 1. Caso as entradas sejam iguais, marcar com x ou y (a ou b). Caso contrário, rejeite. Continue até a entrada vazia.
6. Mova o cabeçote uma posição à esquerda na fita 1 (neste momento, o cabeçote da fita 1 irá apontar para o final da última string) e mantenha o cabeçote 2 onde está (neste ponto, ele está no começo da string reversa).

7. Percorra a fita 1 no sentido direita-esquerda e a fita 2 no sentido esquerda-direita. Caso as entradas sejam iguais, continue até atingir uma entrada a ou b na fita 1 (alertando que acabou a string). Atingindo esta marca, aceite a palavra.

As imagens dos testes estão em Imagens/2b A codificação da máquina está em *machines_jflap*

4 Máquina 3a - Blocos

4.1 Enunciado da linguagem

$$L = \{ 0^{2^n} \mid n \geq 0 \}$$

4.2 Descrição do algoritmo

1. Inicialmente, verifica-se se a palavra é vazia ou não. Se for, vai para o estado de rejeição. Caso contrário se for um 0, escreve x e vai para o próximo estado.
2. No próximo estado, ficará procurando por 0 ou por palavra vazia. Caso seja uma palavra vazia, significa que entrou num estado de aceitação, então aceita. Se encontrar x apenas avança na fita e se encontrar um 0 escreve x na fita e vai para o próximo estado.
3. Ficaré nesse estado até encontrar um a palavra vazia ou um 0. Em caso de x, apenas avança na fita. Se encontrar uma palavra vazia, retrocede na fita e troca de estado. Caso encontre um 0, avança na fita e vai pro outro estado.
4. Se encontrar um 0, escreve x na fita e volta para o estado anterior. Se encontrar um x, apenas avança na fita e caso encontre uma palavra vazia significa que a palavra ao todo tem um número diferente de 0^{2^n} , então vai para o estado de rejeição.
5. Ficaré voltando na fita, para cada x e 0 na fita. Quando encontrar uma palavra vazia, avança e volta para o segundo estado da máquina As imagens dos testes estão em /Imagens/3a

5 Máquina 3b - Blocos

5.1 Enunciado da linguagem

$$L = \{ \text{Um somador binário} \}$$

5.2 Descrição do algoritmo

1. A máquina possui 4 fitas. Duas para os operandos, uma para o "carry" e uma quarta para o resultado. Inicialmente a máquina percorrerá as duas fitas com os operandos até chegar o fim. A máquina aceita operandos com números de bits. Quando ela encontrar a palavra vazia nos dois operandos, significa que chegou no fim das duas, então muda de estado.
2. A máquina compara a entrada de cada palavra dos operandos e com a entrada do "carry" e escreve o resultado baseado nessa entrada (usando um comportamento de um somador obviamente). O cabeçote do "carry" está sempre no mesmo lugar, visto que não há necessidade de mudá-lo. Quando a máquina encontrar a palavra vazia nos dois operandos vai para o estado de aceitação. Por convenção, ela sempre escreve o "carry" no bit mais significativo, independentemente de qual valor estava no carry. As imagens dos testes estão em /Imagens/3b