

Felipe de Campos Santos (17200441)

Trabalho de Máquinas de Turing - Teoria da computação (INE5415)

Outubro de 2020

1. Implemente Máquinas de Turing com fita única para computar as seguintes linguagens:

(a) (1,0pt)  $L = \{a^i b^j c^k \mid i, j, k \in \mathbb{N} \text{ e } i \times j = k\}$

(b) (1,0pt)  $L = \{0^{2^n} \mid n \geq 0\}$

a) \*\*OBS: foi usado o símbolo ">" para representar inicio de fita

a) Se a palavra for vazia, **aceite**

b) Varra todos os a's

i) Se depois de todos os a's vier vazio, **aceite**

ii) Se depois de todos os a's vier um c, **rejeite**.

iii) Se depois de todos os a's vier um b, continue.

c) Varra todos os b's

i) Se não existiu nenhum a anterior e depois de todos os b's vier vazio, **aceite**

ii) Se não existiu nenhum a anterior e depois de todos os b's vier um c, **rejeite**

iii) Se existia a's antes dos b's, continue

d) Varra todos os c's

i) se a palavra começou com c, **rejeite**

e) Volte até encontrar um b e marque-o com B

f) Avance todos os b's até encontrar um c e marque-o com C

i) se a depois deste C acabar a palavra, **aceite**

ii) se tiver mais um c, retorne até encontrar um b

iii) se encontrar um b, volte para o passo e

iv) se não, avance

g) volte todos os B's até encontrar um a

h) troque todos os B's por b's

i) retorne até encontrar o a do passo g e o troque por A

j) avance todos os a's até encontrar um b

k) troque o b por B e avance até encontrar um c

i) caso não encontre nenhum c, então  $a \cdot b > c$ , **rejeite**

ii) caso encontre, volte ao passo f

b)

a) se for vazio, **rejeite**

b) (a partir daqui vamos diminuir a palavra pela metade e ver se sobra par) marque o primeiro 0 com vazio

i) se depois disso vier vazio, **aceite** (para  $n=0$ , temos  $2^0 = 1$ , logo apenas um zero)

ii) se tiver outro 0, marque ele com 'z' e siga pra direita.

c) daqui, marcaremos com z um 0 não e um 0 sim (diminuindo os zeros restantes pela metade)

- d) Depois de marcar, a maquina retorna ao começo da fita e conta o numero de z.
- i) Se for um numero impar, **rejeite**
  - ii) Se for um número par, volte ao passo c

## 2. Implemente Máquinas de Turing Multifitas para computar as seguintes linguagens:

- (a) (1,5pt)  $L = \{ww^Rw^Rw^R \mid w \in \{0, 1\}^*\}$  ( $w^R$  é o reverso da cadeia  $w$ )
- (b) (1,5pt)  $L = \{\#x_1\#x_2\#\dots\#x_n \mid x_i \in 0, 1^* \text{ e } \exists x_i \neq x_j \text{ para cada } i \neq j\}$

a)

- a) Copie o valor da entrada para a segunda fita
- b) Na segunda fita, percorra a palavra trocando os valores por K, do começo ao fim da fita (percorrendo à direita até encontrar um vazio ou um #) da seguinte maneira (0,i,1,i-1,2,i-2,...) até que sobre apenas um valor diferente de K
- c) Coloque um # a esquerda desse valor
- d) À direita de #, copie os valores da celula i da primeira fita para a celula i+1 da segunda fita (valores deslocados uma celula, pois foi acrescentado o #).
- e) Volte ao começo da fita
- f) copie os valores da celula i da primeira fita para a celula i da segunda fita, até na segunda fita encontrar #.
- g) Temos a palavra dividida ao meio
- h) Volte ao passo B. Teremos a palavra dividida em 4 partes iguais
- i) Pegue o primeiro simbolo e copie ele para a terceira fita e troque o na segunda fita para K
- j) Ande até encontrar o segundo #, K ou vazio
- k) retorne uma celula a esquerda e compare com o simbolo da terceira fita
  - i) se for igual, apague ele da terceira fita e troque o simbolo sob o cabeçote da segunda fita para K
  - ii) Se for diferente, **rejeite**
- l) Retorne à esquerda até encontrar K
- m) volte ao passo i
- n) Se o valor da esquerda for #, terminamos essa metade
- o) ande à direita
  - i) se achar vazio, **aceite**(a palavra terminou)
  - ii) se achar #, volte ao passo i (estaremos na segunda metade da palavra)

b)

- a) Se a entrada for vazia ou tiver apenas um #, **aceite**. Se não, siga
- b) Ande a direita até encontrar um #
- c) copie os simbolos que vem depois desse # para a segunda fita, e na primeira troque-os por k
- d) ao encontrar mais um #, pare de copiar
- e) mova a direita
- f) retorne ao inicio da segunda fita
- g) ande a direita comparando o que esta na primeira e na segunda fita
  - i) se algum simbolo for diferente, **aceite**
  - ii) caso contrário
    - 1) se chegar ao final da primeira palavra (ler vazio na primeira fita), **rejeite**
    - 2) se nao, continue
- h) pare ao encontrar o proximo # e volte ao passo e

3. Implemente Máquinas de Turing a sua escolha para computar os seguinte problemas:

- (a) (2,5pt) A série de Fibonacci. A máquina recebe como entrada uma sequência de símbolos que representa  $n$ . Ao término, deve constar na fita uma sequência de símbolos que indica o valor do  $n$ -ésimo termo, ou seja  $Fibonacci(n)$ .
- (b) (2,5pt) O algoritmo de Euclides para o Máximo Divisor Comum. A máquina recebe como entrada uma sequência de símbolos representando  $n$  e  $m$ . Ao término, a fita deve conter o  $MDC(n, m)$ .

a)

- a) a maquina recebe sequencias de a's pro index desejado de fibonacci
- b) pros 2 primeiros a's, apenas coloca um b na terceira fita, que vai ser utilizada para controle de passo junto com a segunda, e um z na quarta fita, que vai ser utilizada para resultado ( $fib(1)=fib(2)=1$ ), base do algoritmo)
- c) para os seguintes a's, é somada a segunda fita com a terceira usando B's na terceira fita para para controle do ultimo numero somado (valor da segunda fita). O valor dessa é passado pra quarta fita e a terceira fita fica com o ultimo valor somado
- d) ao retornar, na segunda fita é colocada o passo anterior (terceira fita - B's = b's), assim guardando o número anterior.
- e) Os passos são repetidos até que todos os a's estejam trocados por k's
- f) o resultado estará na quarta fita

b)

- a) a máquina recebe dois números, um em cada fita
- b) é percorrido os dois simultaneamente até que se ache vazio
  - i) se achar vazio ao mesmo tempo nos dois, a maquina **aceite**(o MDC é o número que sobrou nas fitas. Caso desde o início eles sejam iguais, no primeiro fim de leitura da palavra já é retornado os proprios numeros)
  - ii) se achar vazio em cima ou embaixo antes, algum dos números é menor. A partir daqui existem dois caminhos que fazem a mesma coisa, a diferença é que um faz na fita de baixo e o outro faz na de cima

- c) a máquina, após chegar ao final das palavras, retorna subtraindo o menor número do maior.
- d) volta ao passo b