

Projecto de Teoria da Computação

0,5+1,0+1,0+1,5 valores

IST, LEIC – 15 de Abril de 2021

Considere a seguinte representação alternativa de máquinas de Turing (com 1 fita de memória bidireccional e movimentos-R,L,S), que utiliza um alfabeto com 18 símbolos $\Sigma = \{Q, A, Y, N, S, L, R, 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, ;\}$ e adopta as seguintes convenções:

- cada máquina é codificada pela sua lista de transições, separadas por ponto-e-vírgula, no formato

$\text{trans}_1; \text{trans}_2; \dots; \text{trans}_z$

- cada transição tem o formato

$\text{estado}_A \text{símbolo}_A \text{estado}_B \text{símbolo}_B \text{movimento}$

onde estado_A é o estado de partida da transição, estado_B o estado de chegada, símbolo_A é o símbolo lido pela transição, símbolo_B o símbolo escrito pela transição, e movimento o movimento executado pela transição

- cada estado de controlo da máquina é representado por uma palavra da forma

$Qd_1 \dots d_K$

onde cada símbolo $d_i \in \{0, 1, 2, 3, 4, 5, 6, 7, 8, 9\}$ (o valor de K , inteiro positivo, é igual para todos os estados da máquina); o estado inicial da máquina é $Q0^K$; os estados de aceitação e rejeição da máquina são representados, respectivamente, por Y^{K+1} e N^{K+1}

- cada símbolo do alfabeto de trabalho da máquina é representado por uma palavra da forma

$A d_1 \dots d_T$

onde cada símbolo $d_i \in \{0, 1, 2, 3, 4, 5, 6, 7, 8, 9\}$ (o valor de T , inteiro positivo, é igual para todos os símbolos do alfabeto); o símbolo $A0^T$ corresponde ao espaço em branco

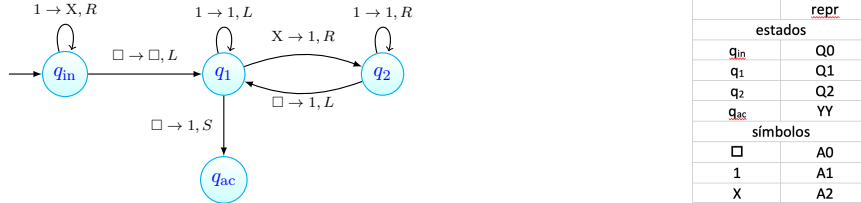
- cada movimento é representado por um dos símbolos R, L ou S.

Note que todos os estados de uma máquina são representados por palavras com o mesmo comprimento $(K+1)$. Analogamente, todos os símbolos de uma máquina são representados por palavras com o mesmo comprimento $(T+1)$. Obviamente, os valores de K e T podem variar de máquina para máquina (uma máquina com mais de 10 estados de controlo precisará de usar $K>1$, tal como uma máquina com alfabeto de trabalho com mais de 10 símbolos precisará de usar $T>1$).

Por exemplo, a palavra

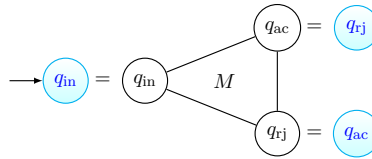
Q0A1Q0A2R;Q0A0Q1A0L;Q1A1Q1A1L;Q1A2Q2A1R;Q1A0YYA1S;Q2A1Q2A1R;Q2A0Q1A1L

denota a máquina desenhada abaixo, adoptando a representação indicada à direita.

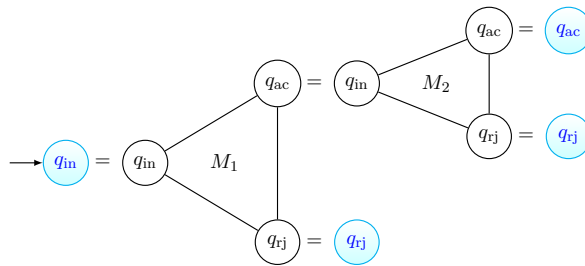


Pretende-se definir quatro máquinas de Turing (que podem ser bidireccionais e multifita, se conveniente) para manipular este tipo de representações de máquinas:

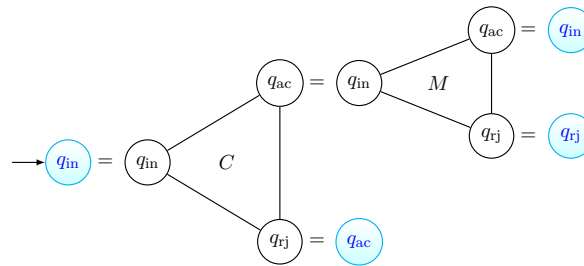
- (a) uma máquina (**complem.txt**) que transforma a representação de uma máquina de Turing M na representação da máquina de Turing complementar $comp(M)$, que resulta de nela trocar os estados de aceitação e de rejeição (na linha da Proposição 3.3 do Capítulo 3 das notas de apoio da disciplina), como se ilustra



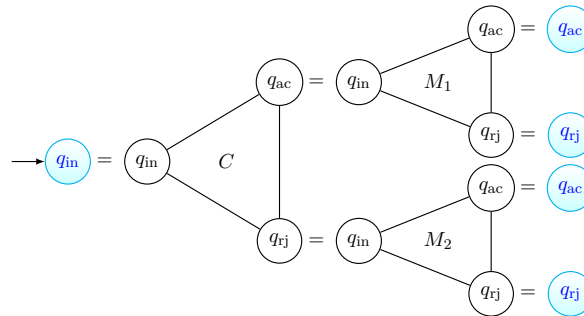
- (b) uma máquina (**compor.txt**) que dada a representação de duas máquinas de Turing (no formato $M_1\$M_2$) calcula a representação da máquina de Turing ($M_1; M_2$) que calcula a função composta (cumprindo os requisitos do Exercício 6.2 do Capítulo 2 das notas de apoio da disciplina), construída como se ilustra



- (c) uma máquina (**while.txt**) que dada a representação de duas máquinas de Turing (no formato $C\$M$) calcula a representação da máquina de Turing (**WHILE** C **DO** M), que executa M transformando sucessivamente o *input* enquanto C aceitar, construída como se ilustra



- (d) uma máquina (`ifthenelse.txt`) que dada a representação de três máquinas de Turing (no formato $C\$M_1\M_2) calcula a representação da máquina de Turing (`IF C THEN M_1 ELSE M_2`), que executa M_1 ou M_2 sobre o *input* consoante C o aceite ou rejeite, construída como se ilustra



Nota: nas alíneas (b,c,d) o comportamento pretendido assume que as representações das várias máquinas dadas como *input* utilizam o mesmo alfabeto de trabalho, codificado exactamente da mesma forma; nas alíneas (c,d) o comportamento pretendido assume que a máquina C , que serve de condição Booleana, devolva o *input* inalterado.

Instruções:

- o trabalho deve ser realizado em grupos de 3 alunos, devidamente inscritos no fénix (qualquer excepção necessita de autorização dos docentes)
- cada grupo deve entregar quatro ficheiros de texto, contendo o código de cada uma das máquinas: `complem.txt`, `compor.txt`, `while.txt` e `ifthenelse.txt`
- em todos os casos, o código deve ser executável directamente pelo emulador disponibilizado [aqui](#)
- pode e deve testar as suas máquinas usando o emulador, e tirando partido da máquina universal para esta representação disponibilizada no ficheiro `maquniv.txt` anexo
- para efeitos de teste, pode considerar as representações de máquinas fornecidas no ficheiro `exemplos.txt` anexo, que tal como o exemplo acima manipulam números naturais representados em unário
- os trabalhos submetem-se no fénix até às 23h59m de 25 de Maio de 2021.