Theory of Computer Science

G. Röger
S. Eriksson
Spring Term 2023

University of Basel
Computer Science

Exercise Sheet 6

Due: Wednesday, April 19, 2023

Due to the Easter break, this exercise sheet covers 3 lectures: April 3, April 5 and April 12. It is thus worth 15 points and you have two weeks to solve it (including Easter break).

Exercise 6.1 (PDAs; 1 point)

This exercise is a question that was asked on the lecture slides.
Assume you want to have a possible transition from state q to state q' in your PDA that processes symbol c from the input word, can only be taken if the top stack symbol is **A**, does not pop **A** off the stack, and pushes **B**.
What problem do you encounter? How can you work around it?

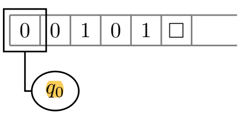
Exercise 6.2 (Turing Machines; 2+2+2 points)

Show how the following steps used in the proof of multi-tape Turing machines can be done by a Turing machine with a single tape.

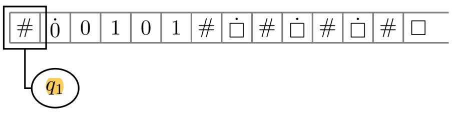
The examples in the exercises are just meant for illustration. Your solution should work for arbitrary multi-tape TMs and arbitrary inputs. Describe the functionality in natural language (the question is not about a concrete TM). The number of tapes k is known during the construction but the length of the input is not.

(a) On input of word $w = w_1 \dots w_n$, initialize the tape with $\#\dot{w}_1 w_2 \dots w_n \#\dot{\square}\#\dot{\square}\# \dots \#$.

For example, in this situation

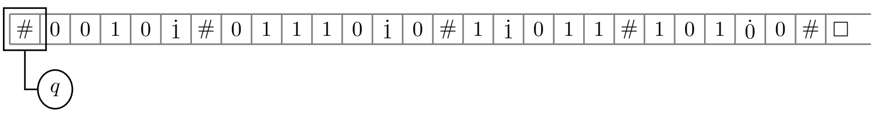


this one should be reached:

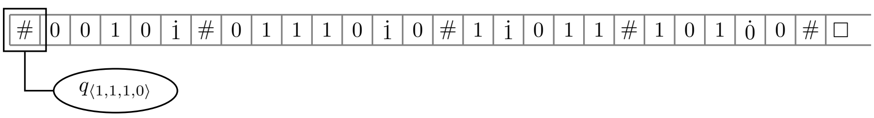


(b) Assume the read-write head is on the first position of the tape, the machine is in state q, and $a_1, \dots, a_k$ are the symbols under the virtual heads (i.e., a_i is the symbol with the dot on virtual tape i for all $1 \le i \le k$). Reach the situation where the head is again on the first position of the tape but the machine is in state $q_{\langle a_1, \dots, a_k \rangle}$

For example, in this situation

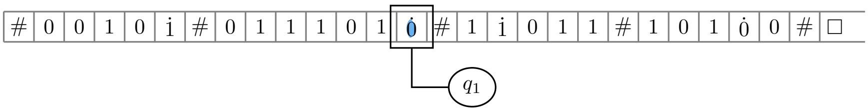


this one should be reached:

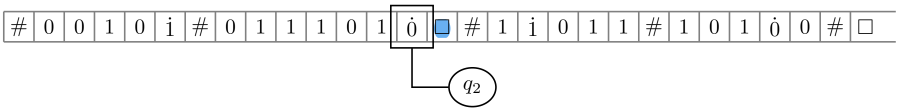


(c) Insert a □ at the current position by shifting the complete tape content right of the current position one step to the right.

For example, in this situation



this one should be reached:



Exercise 6.3 (Closure Properties of Type 0 Languages; 1+1 point)

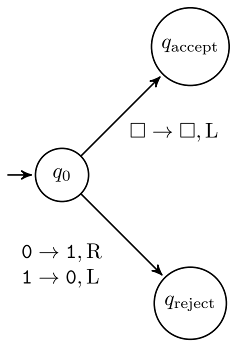
Explain on a high level why the following ideas for showing closure properties do not work:

(a) Given a DTM M that recognizes language L, we can build a DTM M' that recognizes $\bar{L}$ by inverting the accept and reject states of M.

(b) Given NTMs M_1 and M_2 of languages L_1 and L_2, we can build a NTM of language $L_1 \cap L_2$ by building a "product NTM" $M_{1,2}$ such that for all transitions pairs $(q_i, x) \to (q_j, y, D)$ from M_1 and $(q_{i'}, x) \to (q_{j'}, y, D)$ from M_2 there is a transition $(q_{(i,i')}, x) \to (q_{(j,j')}, y, D)$ in $M_{1,2}$.

Exercise 6.4 (Encoding of Turing machines; 2+2 points)

(a) Encode the following Turing machine as a word over the alphabet $\Sigma = \{0, 1\}$. Provide an encoding over $\{0, 1, \#\}$ as intermediate solution.



(b) Specify the state diagram of Turing machine M_w, encoded by the following word w

$$w = 11110011001100110011011111100110111011101110111110011010011010011001101$$

Exercise 6.5 (Decidability; 1+1 points)

Which of the following statements are true? Briefly justify your answers with one or two sentences each. You can use all results from the lecture.

(a) All type-2 languages are Turing-decidable.

(b) If a language L is Turing-decidable, then there is a grammar that generates $\bar{L}$.