Gruppe	Hemmecke (10:15)	Hemmecke (11:00)		Popov
Name		Matrikel		SKZ

Klausur 1

Berechenbarkeit und Komplexität

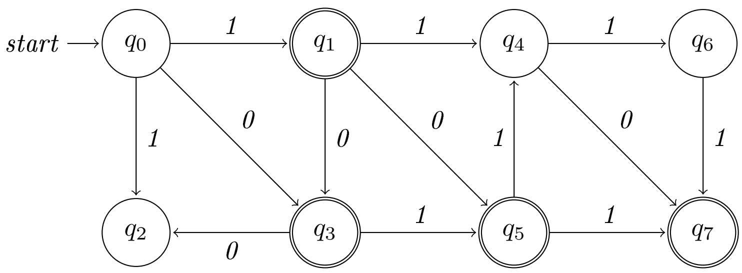
30. November 2018

Part 1 NFSM2018

Let N be the nondeterministic finite state machine

$$(\{q_0, q_1, q_2, q_3, q_4, q_5, q_6, q_7\}, \{0, 1\}, \nu, \{q_0\}, \{q_1, q_3, q_5, q_7\}),$$

whose transition function ν is given below.



1		no

Is $1001100111 \in L(N)$?

The sequence is not defined by the transition.

2	yes	

Is $101111 \in L(N)$?

Follow the states $q_0, q_1, q_3, q_5, q_4, q_6, q_7$.

3	yes	

Is $L(N)$ finite?

4	yes	

Does there exist a regular expression r such that $L(r) = \overline{L(N)} = \{0,1\}^* \setminus L(N)$?

$L(N)$ is regular and so is its complement.

5	yes	

Is $\overline{L(N)}$ recursively enumerable?

$L(N)$ is regular. Hence, $\overline{L(N)}$ is regular, and thus also recursively enumerable.

6	yes	

Is there a deterministic finite state machine M with less than 2018 states such that $L(M) = L(N)$?

According to the subset construction, there must be a DFSM with at most $2^8 = 256$ states.

7	yes	

Is there an enumerator Turing machine G such that $Gen(G) = L(N)$?

8	yes	

Does there exists a deterministic finite state machine D such that $L(D) = L(N) \circ \overline{L(N)}$?

$L(N)$ and $\overline{L(N)}$ are both regular. Concatenation of two regular languages gives a regular language.

Part 2 Computable2018

Let M_1 be a Turing machine such that it accepts a word, if and only if it is a palindrome. A palindrome is a word that can be read the same way from either

direction, left-to-right or right-to-left. For example, wow, solos, level, kayak, ABBA, otto *and* redder *are palindromes.*
Let M_2 be a Turing machine such that it accepts a word, if and only if it is a tautonym. A tautonym is a word or a name made up of two identical parts, such as soso, tomtom, BadenBaden *or* PagoPago.
We assume that the alphabets of M_1 and M_2 coincide.

9	yes		*Is $L(M_1) \cap L(M_2)$ recursively enumerable?*
10	yes		*Is $L(M_1) \cap L(M_2)$ recursive?*
11		no	*Is $L(M_1) \cap L(M_2)$ finite?*

There can be arbitrarily large words being palindromes and tautonyms at the same time.

12		no	*Let L be a recursively enumerable language. Can it be concluded that $L(M_1) \cap L(M_2) \cap L$ is recursive?*

Intersection of recursive and recursively enumerable languages is recursively enumerable but not necessarily recursive.

13		no	*Is every μ-recursive function also a primitive recursive function?*
14		no	*Does there exist a μ-recursive function that is not WHILE computable?*
15	yes		*Is every primitive recursive function also Turing-computable?*

Part 3 *Pumping2018*
Let

$$L_1 = \left\{ a^m b^n a^{2m} \,\middle|\, m, n \in \mathbb{N}, m < 2018 \right\},$$
$$L_2 = \left\{ a^m b^n a^{2m} \,\middle|\, m, n \in \mathbb{N}, n < 2018 \right\}.$$

16	yes		*Is there a regular expression r such that $L(r) = L_1$?*

$r = b^* + ab^*aa + aab^*aaaa + \cdots + a^{2017}b^*a^{4034}$

17		no	*Is there a deterministic finite state machine M such that $L(M) = L_2$?*

L_2 is not regular.

18	yes		*Is there an enumerator Turing machine G such that $Gen(G) = L_1$?*
19	yes		*Is there an Turing machine M such that $L(M) = L_1 \cup L_2$?*
20	yes		*Is there an deterministic finite state machine D such that $L(D) = L_1 \cap L_2$?*

The language $L_1 \cap L_2$ is finite and thus regular.

Part 4 *WhileLoop2018*
Let T_1 and T_2 be two Turing machines. Assume that T_1 and T_2 compute partial functions $t_1, t_2 : \mathbb{N} \to \mathbb{N}$, respectively, and that t_1 is a total function whereas t_2 is undefined for at least one input $i \in \mathbb{N}$. (We assume that a natural number n is encoded on the tape as a string of n letters 0.)

21		no

Can it be concluded that t_1 is LOOP-computable?

The Ackermann function ack is a total function that is not primitive recursive. Hence, if T_1 is the Turing machine that computes $t_1(n) = \mathrm{ack(n, n)}$, then we can assume that T_1 halts on every input. However, since t_1 is not primitive recursive, there cannot be a corresponding LOOP-program.

22	yes	

Is there a WHILE-program that computes t_2?

Every Turing computable function can be simulated by a WHILE-program.

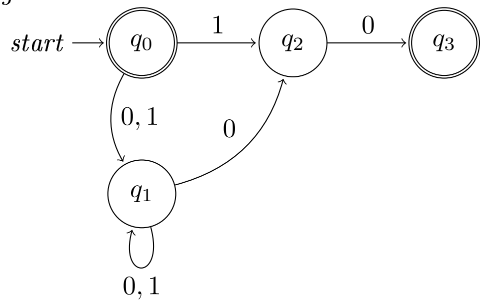
23	yes	

Is the composition $t_1 \circ t_2$ a μ-recursive function?
Hint: $(t_1 \circ t_2)(x) = t_1(t_2(x))$, *if t_2 is defined on x and undefined otherwise.*

Part 5 *Open2018*

((2 points))

Let $N = (Q, \Sigma, \delta, q_0, F)$ be a nondeterministic finite state machine with $Q = \{q_0, q_1, q_2, q_3\}$, $\Sigma = \{0, 1\}$, $S = \{q_0\}$, $F = \{q_0, q_3\}$, and transition function δ as given below.



1. *Let X_i denote the regular expression for the language accepted by N when starting in state q_i.*

   Write down an equation system for $X_0, \ldots, X_3$.

2. *Give a regular expression r such that $L(r) = L(N)$ (you may apply Arden's Lemma to the result of 1).*

$$
\begin{aligned}
X_0 &= (0+1)X_1 + 1X_2 + \varepsilon \\
X_1 &= (0+1)X_1 + 0X_2 \\
X_2 &= 0X_3 \\
X_3 &= \varepsilon \\
r &= \varepsilon + 10 + (0+1)(0+1)^*00
\end{aligned}
$$