Mälardalen University
School of Innovation, Design and Engineering

TEN1 DVA337 2015

Formal languages, automata, and theory of computation

Thursday, November 5, 14:10 - 18:30
Teacher: Daniel Hedin, phone 021-107052

The exam has a total of 40 points and consists of 3 pages. No aids are allowed. Answers must be given in English and should be clearly justified.

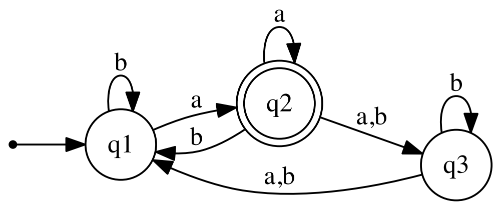
1. Regular languages (14 p)

a) Floating point numbers are built by an optional sign followed by digits separated by a decimal point and end with an optional exponent part. Examples of floating point numbers are

```
1.35e+10  +2.54  -5.2  245.3E-2
```

Write a regular expression that recognizes floating point numbers. (2 p)

b) Convert the following NFA to an equivalent minimal DFA. (6 p)



c) State the pumping lemma for regular languages and use it to prove that $L = \{a^n b^n \mid n \geq 0\}$ is not regular. (4 p)

d) Consider the problem of nested C-like comments. A comment is well-formed if the number of starting symbols `/*` matches the number of ending symbols `*/` and in any given prefix the number of starting symbols is greater than or equal to the number of ending symbols. To illustrate consider the following examples of well-formed nested comments

```
/* Well-formed comment */
/* Also /* a well-formed nested comment */ !! */
```

and the following examples of malformed nested comments

```
/* Not a */ well-formed comment */
/* Also not /* a well-formed comment  */
```

Give a convincing argument that it is not possible to create a DFA that accepts precisely the well-formed nested comments. (2 p)

2. Context-free languages (14 p)

a) Explain what it means for a grammar to be ambiguous. (2 p)

b) Consider the following small grammar for an expression language

$$E ::= E + E \mid E * E \mid NUM$$

where NUM represents numbers. Show that this grammar is ambiguous. (2 p)

c) Rewrite the above grammar to be unambiguous and show by example that your grammar follows the standard precedence rules for addition and multiplication. (2 p)

d) Consider the following grammar of well-formed nested comments

$$\begin{array}{lcl} S & ::= & \lambda \mid / * A * / \\ A & ::= & \lambda \mid aA \mid SA \end{array}$$

where a, $*$, and $/$ are terminals. Create a deterministic pushdown automaton that recognizes the above grammar, and show the sequence of instantaneous description corresponding to running the automaton on 1) `/*a/**/*/` and 2) `/**/a` (6 p)

e) Do deterministic pushdown automata have the same computational power as non-deterministic pushdown automata? Give a convincing argument for or against. (2 p)

3. Restriction-free languages and theory of computation (12 p)

a) Consider the Turing machine described by

$$M = (\{q_1, q_2, q_2, q_3, q_4, q_5, q_6, q_7\}, \{a, b\}, \{a, b, \Box\}, \delta, q_1, \Box, \{q_7\})$$

with δ defined as follows

$$\begin{array}{lclclclclcl}
\delta(q_1,a) & = & (q_2,\Box,R) & \delta(q_2,a) & = & (q_2,a,R) & \delta(q_3,a) & = & (q_3,a,R) \\
\delta(q_1,b) & = & (q_3,\Box,R) & \delta(q_2,b) & = & (q_2,b,R) & \delta(q_3,b) & = & (q_3,b,R) \\
\delta(q_1,\Box) & = & (q_7,\Box,L) & \delta(q_2,\Box) & = & (q_4,\Box,L) & \delta(q_3,\Box) & = & (q_5,\Box,L) \\
 & & & & & & & & \\
\delta(q_4,a) & = & (q_6,\Box,L) & \delta(q_5,b) & = & (q_6,\Box,L) & \delta(q_6,a) & = & (q_6,a,L) \\
 & & & & & & \delta(q_6,b) & = & (q_6,b,L) \\
 & & & & & & \delta(q_6,\Box) & = & (q_1,\Box,R)
\end{array}$$

1. Draw the transition graph of M. (2 p)
2. What language does M accept? (2 p)
3. Select a string $w \in \{a, b\}^4$ that is accepted by M and show the execution of M on w as a sequence of instantaneous descriptions. (2 p)

b) Explain the concept of reduction proof in the context of theory of computation, i.e., proving undecidability by reducing one problem to another. (2 p)

c) Recall the state-entry problem, i.e., given a Turing machine $M = (Q, \Sigma, \Gamma, \delta, q_0, \Box, F)$ and any $q \in Q$, and $w \in \Sigma^+$, decide whether or not the state q is entered by M when it is applied to w. Prove that the state-entry problem is undecidable. (4 p)