Tutorial 1

DA4005 Algorithms and complexity

TA: Anna Lindeberg

Tutorials

- Tutorials are for additional examples, exercises, questions, etc.

Tutorials

- Tutorials are for additional examples, exercises, questions, etc.
- For some tutorials, we'll present example solutions to the hand-ins as well.

Tutorials

- Tutorials are for additional examples, exercises, questions, etc.
- For some tutorials, we'll present example solutions to the hand-ins as well.
- Suggestions are welcome! Contact: `anna.lindeberg@math.su.se`

Recap Turing machines

Turing machines are an abstraction of a computing machine (computer). The TM operates on an infinite piece of tape with symbols (which initially contains the input). Formally:

$$T = (Q, \Sigma, \delta : Q \times \Sigma \to Q \times \Sigma \times \{L, R\}, q_0, F \subseteq Q)$$

Recap Turing machines

Turing machines are an abstraction of a computing machine (computer). The TM operates on an infinite piece of tape with symbols (which initially contains the input). Formally:

$$T = (Q, \Sigma, \delta : Q \times \Sigma \to Q \times \Sigma \times \{L, R\}, q_0, F \subseteq Q)$$

$Q \to$ states of the machine, $q_0 \in Q$

Recap Turing machines

Turing machines are an abstraction of a computing machine (computer). The TM operates on an infinite piece of tape with symbols (which initially contains the input). Formally:

$$T = (Q, \Sigma, \delta : Q \times \Sigma \to Q \times \Sigma \times \{L, R\}, q_0, F \subseteq Q)$$

$Q \to$ states of the machine, $q_0 \in Q$
$\Sigma \to$ input alphabet

Recap Turing machines

Turing machines are an abstraction of a computing machine (computer). The TM operates on an infinite piece of tape with symbols (which initially contains the input). Formally:

$$T = (Q, \Sigma, \delta : Q \times \Sigma \to Q \times \Sigma \times \{L, R\}, q_0, F \subseteq Q)$$

$Q \to$ states of the machine, $q_0 \in Q$
$\Sigma \to$ input alphabet
$\delta \to$ transition func. "If we're in state q and head reads symbol e, transition to state q'. Write e' and move L/R"

Recap Turing machines

Turing machines are an abstraction of a computing machine (computer). The TM operates on an infinite piece of tape with symbols (which initially contains the input). Formally:

$$T = (Q, \Sigma, \delta : Q \times \Sigma \rightarrow Q \times \Sigma \times \{L, R\}, q_0, F \subseteq Q)$$

$Q \rightarrow$ states of the machine, $q_0 \in Q$
$\Sigma \rightarrow$ input alphabet
$\delta \rightarrow$ transition func. "If we're in state q and head reads symbol e, transition to state q'. Write e' and move L/R"
$q_0 \rightarrow$ start state

Recap Turing machines

Turing machines are an abstraction of a computing machine (computer). The TM operates on an infinite piece of tape with symbols (which initially contains the input). Formally:

$$T = (Q, \Sigma, \delta : Q \times \Sigma \to Q \times \Sigma \times \{L, R\}, q_0, F \subseteq Q)$$

$Q \to$ states of the machine, $q_0 \in Q$
$\Sigma \to$ input alphabet
$\delta \to$ transition func. "If we're in state q and head reads symbol e, transition to state q'. Write e' and move L/R"
$q_0 \to$ start state
$F \to$ accept states on wich T terminates

Recap Turing machines

Turing machines are an abstraction of a computing machine (computer). The TM operates on an infinite piece of tape with symbols (which initially contains the input). Formally:

$$T = (Q, \Sigma, \delta : Q \times \Sigma \to Q \times \Sigma \times \{L, R\}, q_0, F \subseteq Q)$$

$Q \to$ states of the machine, $q_0 \in Q$
$\Sigma \to$ input alphabet
$\delta \to$ transition func. "If we're in state q and head reads symbol e, transition to state q'. Write e' and move L/R"
$q_0 \to$ start state
$F \to$ accept states on wich T terminates

To specify a TM, must give Q, Σ, δ, q_0 and F. But this description is hard to visualize!

Working with Turing machines

Consider the TM with $Q = \{q_0, q_1, q_2, q_{\texttt{OK}}\}$, $\Sigma = \{a, b\}$ and $F = \{q_{\texttt{OK}}\}$, where

$$\begin{aligned} \delta\left(q_0, a\right) &= \left(q_1, a, R\right) \\ \delta\left(q_1, a\right) &= \left(q_2, a, R\right) \\ \delta\left(q_2, a\right) &= \left(q_{\texttt{OK}}, a, R\right) \\ \delta(*, b) &= (*, b, R),\ * \in Q \end{aligned}$$

- Draw the simplified finite state representation of this TM
- Examplify what happens on the inputs "$abab$" and "$abaab$"
- What language (=set of strings) does T accept?

Other exercises for Turing machines

1. Provide a TM that takes an integer as input, provided in binary (base 2 representation), and multiplies it by 2.
2. Provide a TM with alphabet $\Sigma = \{H, E, L, O\}$ that accepts every input which contains "$HELLO$", rejects all other
3. Provide a TM that takes an integer as input, provided in binary (base 2 representation), and adds 1.

Big-O/Ω/Θ notation

Be careful when you provide time complexity bounds! It is best to be explicit.

Prove, *rigorously*, that $3n^2 + 2n \in \Theta(n^2)$. What are the relevant definitions? Which constants do you pick?