# List of decision problems

## Jens Kristian Refsgaard Nielsen

### 5. april 2019

## 1 The theory

A decision problem is something that takes an input and asks a question, we wish to determine whether it is possible to make a general algorithm that can answer this question.

### 1.1 Reduces to, $P_1 \leq P_2$

We say a decision problem $P_1 = (I_1, Q_1)$ reduces to another problem $P_2 = (I_2, Q_2)$ iff $\exists F : I_1 \to I_2$ such that $\forall i \in I_1 : i$ is a yes instance of $Q_1$ iff $F(i)$ is a yes instance of $Q_2$.

### 1.2 Reduction theorem

Given two problems and a reduction $P_1 \leq P_2$ we get two very important results:
**Decideable** if $P_2$ is decideable then so is $P_1$,
    *"if you live inside something solveable, then you too are solveable"*
**Undecideable** if $P_1$ is undecideable then so is $P_2$,
    *"if something unsolveable lives inside of you, there is no salvation in more data"*

### 1.3 Rice theorem

#### 1.3.1 Language property

#### 1.3.2 Examples

## 2 The list

In order to decide whether something is decideable or not, it is very important to have a good list of results to use for our deductions. What follows are lists of problems handled in the lectures and exercise classes of "Beregnelighed & Logik"at Aarhus university in the spring of 2019.
Each will be of the form: Name I: input Q: question
Notation:

$$
\begin{aligned}
\text{TM} &= \text{set of turing machines} \\
\text{CFG} &= \text{set of context free grammars} \\
\text{REG} &= \text{set of regular languages}
\end{aligned}
$$

## 2.1 Decideable problems

$$\text{I:} \qquad\qquad\qquad\qquad \text{Q:} \qquad\qquad (2.1)$$

$$\text{I:} \qquad\qquad\qquad\qquad \text{Q:} \qquad\qquad (2.2)$$

## 2.2 Undecideable problems

Halting $\qquad$ I: T∈TM,$w \in \Sigma_T$, $\qquad\qquad$ Q: does T halt on $w$? $\qquad$ (2.3)

$$\text{I:} \qquad\qquad\qquad\qquad\qquad \text{Q:} \qquad (2.4)$$