Was ist theoretische Informatik?

- ◇ (methodisch) ein Teilgebiet der Mathematik
- ◇ kein homogener Block, sondern unterschiedlichste Teildisziplinen
- ◇ 'Kerntheorien' (formale Sprachen, Rekursionstheorie, ...) aber insgesamt keine klaren Abgrenzungen, sehr lebendig

Ein Grundsatzblick auf 'Berechenbarkeit'

Was läßt sich ganz allgemein über Programmierbarkeit (Berechenbarkeit) aussagen?

Grenzen der Berechenbarkeit:
Negative Aussagen ('es gibt keinen Algorithmus, sodass ...') verlangen nach mathematisch handhaberen)
Modellen der Berechenbarkeit

Heutigentags hunderte – auch unkonventionelle – Berechenbarkeitsmodelle. (Rewrite-Syteme, DNA-Computing, Quantum-computing, ...)

Historisch und (bis heute) systematisch zentral:
Turingmaschine (Alan Turing, 1936)

Die Church-Turing-These

Church (1936):
Alle berechenbaren Funktionen sind im λ-Kalkül repräsentierbar (und umgekehrt).

Turing (1936):
Alle Algorithmen (berechenbare Funktionen) sind Turingmaschinen-programmierbar.

Satz (Hauptsatz der Algorithmentheorie):
λ-repräsentierbar = TM-berechenbar = Markov-algorithmisch = ...

Gibt es ein (TM-)Programm, dass folgendes Problem für alle Inputs korrekt löst?
M.a.W.: Ist das Problem entscheidbar?

Halteproblem:
Input: ein (Zahlencode für ein TM-)Programm P
Output: 1 . . . falls P (bei Eingabe 'P') terminiert
0 . . . falls P (bei Eingabe 'P') nicht terminiert

Satz:
Das Halteproblem ist unentscheidbar.

Bemerkung: Programme, die als Input Programme erwarten, sind in der Informatik nichts besonderes: Compiler, Interpreter, . . .

Die Unentscheidbarkeit anderer Probleme folgt direkt:
'optimaler Compiler', Äquivalenzproblem, Korrektheitsproblem, . . .
weniger direkt: Hilbert's 10. Problem, logische Gültigkeit, . . .

Beweis der Unlösbarkeit des Halteproblem

Analog zum Cantor'schen Diagonalverfahren

[`Whiteboard ?`]

Die Unentscheidbarkeit anderer wird durch *Reduktion* von bereits als unentscheidbar erkannten Problemen nachgewiesen.

Der **Gödel'sche Unvollständigkeitssatz** folgt ziemlich direkt: 'Nicht alle arithmetischen wahren Sätze sind (in einem Sytem S) beweisbar' bzw.: 'Arithmetik ist nicht axiomatisierbar'

(Es genügt zu sehen, dass sich untscheidbare Probleme — z.B. Hilbert's 10. Problem — in S ausdrücken lässt.)

Komplexitätstheorie

Vergleiche:
Komplexität eines *Algorithmus* — Komplexität eines *Problems*

Unentscheidbarkeitsnachweise sind Beispiele für
komplexitätstheoretische Ergebnisse

Gegeben ein entscheidbares Problem ist es meist wichtig zu fragen:
Wie schnell (wie Speicherplatz-aufwändig, ...) ist ein optimales
Programm? (Im ungünstigsten Fall, im Durchschnitt, ...)

Wie wird Rechenzeit dabei (vernünftig) gemessen?

Folgendes erweist sich als *robust*[a]:
Anzahl der TM-Rechenschritte relativ zur Bit-Länge des Input

[a] gilt erst ab einem bestimmten Komplexitätslevel

Von Problemen zu Problemklassen

Der Einfachheit halber: nur *Entscheidungs*probleme (0/1-Antwort)

P(-TIME):
Probleme, die in polynomieller Rechenzeit lösbar sind.

Beispiele für **P**:
Liste sortiert? $ax + by = c$ lösbar? AL-Formeln auswerten, ...
Graphentheoretische Probleme: zyklisch? zusammenhängend?

NP(-TIME):
Probleme, für die das Verifizieren von Antworten in **P** ist.

Beispiele für **NP**:
$ax^2 + by = c$ lösbar? AL-Formel erfüllbar? (SAT)
Rucksack optimal gepackt? (und ähnliche Optimierungsprobleme)
Graphentheoretische Probleme: k-Clique? 3-färbbar? hamilton'sch?
Travelling Salesman, ...

$\mathbf{P} \stackrel{?}{=} \mathbf{NP}$ — ein 1-Million-Dollar-Problem

`http://www.claymath.org/Millennium_Prize_Problems/`

Die genannten **NP**-Probleme sind alle *vollständig* bezüglich polynomieller Reduktion:

Antwort auf Input a' für Problem B
= Antwort auf Input a für Problem A,
wobei a in polynomieller Zeit in a' übersetzbar ist.

Konsequenz:
Es genügt zu zeigen, dass (k)ein polynomielles Entscheidungsverfahren für, z.B., SAT exisitiert (SAT $\in \mathbf{P}$?)

Die meisten ähnlichen komplexitätstheoretischen Probleme sind ebenfalls ungelöst: z.B., $\mathbf{P} \stackrel{?}{=} \mathbf{P\text{-}SPACE}$

P-SPACE:
Probleme, die mit polynomiellem Speicherplatzaufwand lösbar sind.