

Wolf: Zu TGI1: Was ist in der Informatik möglich? Wo haben Wir da die Grenze gezogen?

ich: Es gibt berechenbare und nicht berechenbare Funktionen.

Wolf: Was heisst berechenbar?

ich: Wenns ne TM für gibt die diese Funktion berechnet.

Wolf: Wan issn ne TM?

ich: Ein 5-Tupel blah blah! Übergangsrelation(!) = nichtdet. und Übergangsfunktion(!) = deterministisch. (Er hat zwar nicht genauer danach gefragt, aber im Zweifel sollte man das auch anders erklären können bzw. den Unterschied zw. Funktion und Relation definieren können).

Wolf: Damit ist der Algorithmus Begriff auch schon mal konkretisiert. Aber TM kann man ja nicht so recht nachbauen. Was kann man denn nun mit nem Computer alles berechnen?

ich: These von Church: TM und RM und Javaprogramm und Pascal-Prgramm sind alle gleichmächtig, berechnen also alle die selbe Klasse von Funktionen.

Wolf: Gibts denn nicht berechenbare Funktionen?

ich: Jupp! Kann man auch ganz leicht zeigen, dass es mehr Funktionen von $\{0,1\} \rightarrow \{0,1\}$ gibt als es TM über $\{0,1\}$ codiert gibt.

Wolf: Jo welche gips da denn so?

ich: Einige Entscheidungsprobleme sind nicht berechenbar. Ein Entscheidungsproblem ist.....! Entscheidbarkeit bedeutet! Zum Bsp. das Halteproblem, das ist folgendes!

Wolf: Beweisen sie dass das Halteprob. nicht berechenbar ist.

ich: Hab einfach den Pseudocode-beweis auch Wikipedia gelernt und den dort angegeben. Ist ganz leicht!

Wolf: Welche nicht berechenb. Probs. gibts noch so?

ich: Satz von Rice. Also das Halteprobl. jedoch noch mit einer speziellen Ausgabe der TM.

Wolf: Und wie beweist man das?

ich: indem man es auch das Halteprobl. reduziert. (diese beweisidee reichte hier völlig)

Wolf: ZuTGI2: Sprachen! Welche kennen Sie da so?

ich: (Hab das gespräch absichtlich zu den regulären gelenkt, da ich die besser konnte) Es gibt die Endlichen Sprachen. Dazu Endliche Automaten ohne Zyklus! Es gibt reguläre Sprachen, dazu endliche Automaten mit Zyklus. Daraus folgt, dass die Endlichen ne Teilmenge der regulären sind. Dann noch die Kontextfreien, Kontextsensitiven, Entscheidbaren und Aufzählbaren. Jeweils die Implikation dazu.

Wolf: Wassn ne Grammatik?

ich: Ein Tupel..... blah!

Wolf: Was wissen wir denn über ne Sprache die eine Gramatik hat?

ich: Dass sie Aufzählbar ist.

Wolf: Wie beweist man das?

ich: mit dem freien Automat.

Wolf: Japp. Aber der nützt uns nix wegen der unendlichen Zustandsmenge.

ich: Jo mann kan auch ne TM bauen die die Grammatik "simuliert".

Wolf: Und wie?

ich: Na indem jeweils ein Ableitungsschritt der Grammatik durch mehrere Schritte der TM simuliert wird.

Wolf: Und die Rückrichtung?

ich: Die wusste ich nicht, aber irgnediwe kann eine Grammatik wohl auch einen TM-Schritt simulieren. Kein Plan!

Wolf: Sie sagten vorhin Automat. Wadis dat?

ich: Auch ein Tupel. blah blah.....

Wolf: und was sind noch so merkmale von regulären Sprachen?

ich: folgende Eigenschaften sind Äquivalent: es ex. netdet. endl. Automat, es ex. det. endl. Automat, es ex. rechtslinere Gramatik, es ex. nur endlich viele Nerode Äquivalenzen, d.h. der minimale Automat des endl. Automaten ist auch endlich. Soll ich das noch genauer erklären?

Wolf: Nö!

ich: Und reg. Sprachen lassen sich durch durch reg. Ausdruck beschreiben.

Wolf: Wassn dat?

ich: Def. von reg. Ausdruck hingeschriebn und erklärt.

Wolf: zu TDP: Was weisst du über operationelle Sematik.

ich: dass man da die Semantik mit math. Funktionen beschreibt.

Wolf: Jo was für ne Funktion ist das?

ich: von "State" in eine neue Funktion und diese Funktion dann wiederrum von "state" ind "state"

Wolf: Jo ist die Partiell oder total?

ich: Die Funktion ist partiell, weil nicht berechnbare Programme (z.B. ne Endloisschleife) keine Semantik hat, demnach ist diese Funktion an diesen Stellen nicht definiert.

Wolf: Schreib mal die Regel für Assign hin.

ich: *schreib* (hihi kurz vorher auswendig gelernt)

Wolf: So jetzt mal die für die hintereinanderausführung, die war ja etwas komisch.

ich: Jo einfach die Mathematische Verkettung in umgekehrter Reihenfolge, weil man ja immer erst die innere Funktion berechnen muss. Soll ich das jetzt echt hinschreiben?

Wolf: Nö das reicht mir so. Und jetzt noch die Für WHILE bitte!

ich: *hinschreib* das mit dem Fixpunkt!

Wolf: Warum ist das ein Fixpunkt?

ich: math. Definition für Fixpunkt hingeschrieben und gesagt: guck das was vorne steht steht da hinten auch und genau so ist es bei der While Schleife. Die wird halt genau einmal abgerollt.

Wolf: richtig

ich: (puh die Erklärung war echt platt, ein glück er hat sie so hingenommen, denn besser konnt ichs ned erklären)

Wolf: raus jetzt hier eh!

ENDE

Ich habe ja auch von anderen was gehört und stelle allgemein fest:

TGI 1: ENTWEDER Berechnbarkeit und Entscheidbarkeit (hatte ich) ODER Komplexität und Reduzierbarkeit (P und NP und so)

TGI 2: ENTWEDER reguläre Sprachen (hatte ich) ODER kontextfreie Sprachen (Kellerautomaten etc.) und zu beiden sollte man auch die Pumping Lemmata und deren Beweise können. Das kam bei mir nur nicht, weil ich so langsam gesprochen hatte.

TDP : ENTWEDER denotatielle ODER operationelle ODER axiomatische Semantik

Mein Ergebnis: "Sehr Gut"