Hinweise:

* Klausur: Freitag, 2. März 2012, $14^{\underline{00}}$ - $16^{\underline{00}}$

* Vorrechnen in den Übungen!

Entscheidbarkeit

Ein Problem heißt **entscheidbar**, falls es ein Verfahren A gibt, d.d.:

* A stoppt nach endlich vielen Schritten bei Eingabe einer *beliebigen* Instanz des Problems
* gibt dann eine Ja-/Nein-Antwort.

Ein Problem heißt **semi-entscheidbar**, falls es ein Verfahren A gibt, d.d.:

* A stoppt nach endlich vielen Schritten nach Eingabe einer *positiven* Instanz
* gibt dann eine Ja-Antwort
* gibt keine (Ja-) Antwort nach Eingabe einer negativen Instanz.

Bspe:

(a) SAT_{PL} („Erfüllbarkeitsproblem der Aussagenlogik")

Gegeben: Eine aussagenlogische Formel φ

Gefragt: Ist φ erfüllbar?

→ entscheidbar (in φ kommen nur endlich viele propositionale Variablen vor; erzeuge also in bel. Reihenfolge alle Belegungen dieser Variablen mit Wahrheitswerten und prüfe jeweils, ob diese φ erfüllen)

(b) $UNSAT_{PL}$

Gegeben: Eine a.-l. Formel φ

Gefragt: Ist φ nicht erfüllbar?

→ entscheidbar (komplementär zu SAT_{PL}; vertausche einfach Antworten in einem Entscheidungsverfahren für SAT_{PL}.

(c) $\underline{VAL}_{PL}$ ("Gültigkeitsproblem")

Gegeben: Eine a.-l. Formel φ.

Gefragt: Ist φ gültig (wahr bei beliebiger Belegung der Variablen)?

Wegen

$$\varphi \text{ gültig} \iff \neg\varphi \text{ nicht erfüllbar}$$

können wir Instanzen (und Antworten eines Entscheidungsverfahrens) ineinander übersetzen: $\underline{VAL}_{PL}$ und $\underline{UNSAT}_{PL}$ sind aufeinander <u>reduzierbar</u> ($\to$ genauer später)

(d) Die entsprechenden Probleme für die Prädikatenlogik 1. Stufe (FOL) sind unentscheidbar.

<u>Aber</u>: Es gibt Semi-Entscheidungsverfahren für $\underline{VAL}_{FOL}$, weil man alle präd.-logisch gültigen Formeln rekursiv aufzählen kann

Es gibt kein Semi-Entscheidungsverfahren für $\underline{SAT}_{FOL}$.

Definition: Eine Sprache $L \subseteq \Sigma^*$ heißt entscheidbar, falls die charakteristische Funktion von L

$$\chi_L : \Sigma^* \longrightarrow \{0,1\}, \quad \chi_L(w) := \begin{cases} 1, & \text{falls } w \in L \\ 0, & \text{sonst} \end{cases}$$

(Turing, WHILE-, ..) berechenbar ist.

L heißt semi-entscheidbar, falls die "halbe" charakteristische Funktion von L

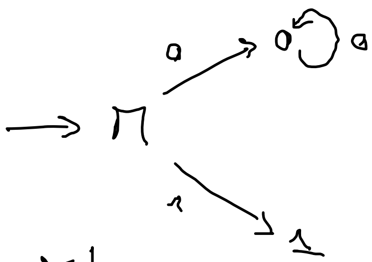
$$\chi_L' : \Sigma^* \nrightarrow \{0,1\}, \quad \chi_L'(w) := \begin{cases} 1, & \text{falls } w \in L \\ \text{undefiniert}, & \text{sonst} \end{cases}$$

berechenbar ist.

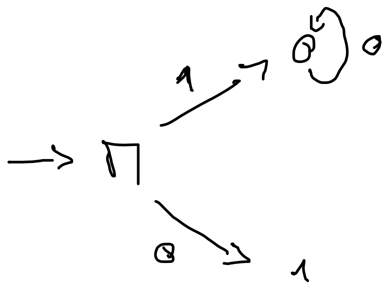
Beachte: χ_L' ist eine partielle Funktion mit Definitionsbereich L.

<u>Satz</u>: Eine Sprache $L \subseteq \Sigma^*$ ist genau dann entscheidbar, wenn sowohl L als auch $\overline{L}$ $(= \Sigma^* \setminus L)$ semi-entscheidbar sind.

<u>Beweis</u>: „$\Rightarrow$" Angenommen L ist entscheidbar. Dann gibt es eine TM M, die χ_L berechnet. Mit M lassen sich TM angeben, die χ'_L bzw. $\chi'_{\overline{L}}$ berechnen:



M'



$\overline{M}'$

„⇐" Angen. L und $\bar{L}$ sind semi-entscheidbar mit TM M' und $\bar{M}'$, die die Funktionen χ'_L bzw. $\chi'_{\bar{L}}$ berechnen (o.E. seien M' und $\bar{M}'$ als 1-Band-TM realisiert).

Konstruiere eine neue TM M als 2-Band-TM wie folgt:

(a) M kopiert zunächst die Eingabe von Band $\underline{1}$ auf Band 2.

(b) M arbeitet danach auf Band $\underline{1}$ wie M' und auf Band 2 wie $\bar{M}'$, d.h. das Eingabewort wird parallel auf beiden Bändern verarbeitet.

(c) Stoppt M' mit Wert $\underline{1}$ auf Band $\underline{1}$, so löscht M Band 2 und stoppt. Stoppt $\bar{M}'$ mit Wert $\underline{1}$ auf Band 2, so löscht M beide Bänder, schreibt 0 auf Band $\underline{1}$ und stoppt.

Da L und $\bar{L}$ semi-entscheidbar sind, stoppt M bei der Eingabe eines beliebigen $w \in L \cup \bar{L} = \Sigma^*$ nach endlich vielen Schritten und gibt 1 oder 0 zurück (f. $w \in L$ / $w \notin L$). Also berechnet M χ_L.

Definition: $L \subseteq \Sigma^*$ heißt **rekursiv aufzählbar**, falls $L \neq \emptyset$ oder falls es eine (links-) totale und berechenbare Funktion $f: \mathbb{N} \to \Sigma^*$ gilt mit

$$L = f(\mathbb{N}) := \{ f(n) : n \in \mathbb{N} \}$$

Es wird nicht gefordert, dass f injektiv ist.

Vorsicht: Begriff "rekursiv aufzählbar" ist zu unterscheiden von:

(a) rekursive Sprache $\cong$ entscheidbare Sprache

(b) abzählbare Sprache: L ist abzählbar, falls L endlich oder abzählbar unendlich ist, d.h. falls gilt $L = \emptyset$ oder es gibt eine surjektive Abbildung $f: \mathbb{N} \to L$, also $f(\mathbb{N}) = L$. f muss aber nicht berechenbar sein. z.B. ist jede Teilsprache einer abzählbaren Sprache abzählbar. Entsprechendes gilt <u>nicht</u> für rekursiv aufzählbare Sprachen (z.B. $L = \Sigma^*$).

<u>Satz</u>: Eine Sprache $L \subseteq \Sigma^*$ ist genau dann rekursiv aufzählbar, wenn sie semi-entscheidbar ist.

<u>Beweis</u>: „$\Rightarrow$" Sei o.E. $L \neq \emptyset$ rekursiv aufzählbar und $f: \mathbb{N} \to \Sigma^*$ eine totale und berechenbare Funktion mit $f(\mathbb{N}) = L$

Idee für ein Semi-Entscheidungsverfahren: Für ein gegebenes Wort w zähle der Reihe nach $f(0), f(1), \dots$ auf und prüfe, ob $f(n) = w$ ist.

```
INPUT (w)
FOR n = 0, 1, 2, ... DO
    IF f(n) = w THEN
        RETURN 1
    END
END
```

definierbar als WHILE-Programm terminiert nicht f. bel. Eingaben

Wichtig: $f(n)$ lässt sich berechnen und ist für (jedes) n definiert.

"$\Leftarrow$" Sei nun $L \subseteq \Sigma^*$ semi-entscheidbar, also

$\chi'_L := \Sigma^* \to \{0,1\}$ berechenbar durch eine TM M.

<u>Zu zeigen:</u> Es gibt eine totale und berechenbare Fkt $g: \mathbb{N} \to \Sigma^*$ mit $g(\mathbb{N}) = L$

Problem 1: Wir müssen nat. Zahlen Wörter $w \in L$ zuordnen.

Problem 2: M terminiert ggf. nicht

Bsp: Für $a <_{lex} b <_{lex} c$ hat man folgende Auflistung der Wörter:

$\varepsilon, a, b, c, aa, ab, ac, ba, bb, bc, aaa, aab, \ldots$

Definiere auf Σ^* eine Ordnung wie folgt:

$$w < w' :\Leftrightarrow |w| < |w'| \text{ oder } |w| = |w'| \text{ und } w <_{lexik.} w'$$

Für $w \in \Sigma^*$ sei n_w die Stelle, an der w in dieser Ordnung vorkommt. Für $n \in \mathbb{N}$ sei w_n das n-te Wort in der Ordnung.

Definiere nun eine totale und berechenbare Funktion $g: \mathbb{N} \to \Sigma^*$, für die wir $g(n)$ wie folgt berechnen:

```
INPUT (n)

k := e(n)

ℓ := f(n)

IF M stoppt bei Eingabe von w_k nach
      ℓ Schritten und gibt 1 zurück
THEN   RETURN w_k
ELSE   RETURN w*
END
```

Interpretiere n als Codierung eines Paares natürlicher Zahlen; d.h. $n = c(k,\ell)$, wobei $c : \mathbb{N} \times \mathbb{N} \to \mathbb{N}$ die früher def. Paar-Codierungs-Fktion ist und $e, f : \mathbb{N} \to \mathbb{N}$ die Umkehrungsfunktionen mit $e(c(x,y)) = x$ und $f(c(x,y)) = y$.

$w^* \in L$, beliebiges, fest gewähltes Wort

Das Berechnungsverfahren für $g(n)$ stoppt für jedes $n \in \mathbb{N}$ und gibt nur Wörter aus L zurück. Also $g(\mathbb{N}) \subseteq L$.

Umgekehrt gilt für jedes $w \in L$, dass M angesetzt auf w nach endlich vielen Schritten ℓ stoppt (und 1 zurückgibt). Setze $n := c(n_w, \ell)$. Dann gilt $g(n) = w$ und daher $L \subseteq g(\mathbb{N})$.

$g(n)$ ber. sich so: $k = e(n) = e(c(n_w, \ell)) = n_w$, $\ell = f(n) = f(c(n_w, \ell))$.
Also stoppt M bei Eingabe von $w_k = w_{n_w} = w$ und das Verf. gibt w zurück.

Zusammenfassend sind also folgende Aussagen äquivalent:

(a) L ist semi-entscheidbar. Def

(b) χ'_L ist (Turing-, WHILE-, GOTO-) berechenbar. FS 16

Def leicht

(c) L ist Definitionsbereich einer berechenbaren Funktion.

(d) L ist rekursiv aufzählbar. Def leicht

(e) L ist Wertebereich einer berechenbaren Funktion.

hier gezeigt

FS 15

(f) $L = T(M)$ für eine TM M. FS 14

(g) L ist vom Typ 0

Halteproblem

Gibt es eine TM, die für jede TM M und jede Eingabe w für M entscheidet, ob M anhält oder endlos weiterläuft?

⇒ Nächste Vorlesung