دانشگاه تبریز

دانشکده علوم ریاضی

گروه ریاضی محض

پایان‌نامه

برای دریافت درجه کارشناسی ارشد در رشته‌ی

ریاضی محض، گرایش منطق

عنوان

تصمیم‌ناپذیری منطق مرتبه اول

استاد راهنما

سعید صالحی پورمهر

استاد مشاور

محمد شهریاری

پژوهشگر

شکوفه صادقی بی‌غم

۱۳۹۳

ستایش

ستایش می کنم کسی راکه منّتش عظیم است و نعمتش فراوان؛ و رحمتش بر غضبش پیشی گرفته است. سخن و حکم او تمامّیت یافته و قطعی است؛ خواست او نافذ و برهانش رسا و حکمش بر عدالت است.

ستایش می کنم به سان سپاس آن که معترف به ربوبیّتش و پر خضوع در بندگی اوست. و از گناه خویش (بریده) کنده شده و به توحید او اقرار می نماید. و از وعید و بیم عذابش به خود او پناه می برد. و از درگاه پروردگارش امیدوار آمرزشی است که او را نجات بخشد، در روزی که (انسان را به گرفتاری خویش مشغول و) از بستگان و فرزندانش غافل می سازد.

از او یاری و هدایت می جوییم و به او ایمان داریم و بر او توکّل می کنیم. از ضمیری با اخلاص و یقین، برای او) به توحید) گواهی می دهم و او را به یکتایی می شناسم. یکتاشناسی فردی مؤمن و استوار (در یقین). و او را یگانه می شمارم، یگانه دانستن بنده ای خاضع. نه در پادشاهی خود شریکی دارد و نه در آفرینشش یاوری. برتر از آن است که مشاور و وزیری داشته باشد و منزّه است از داشتن همانند و نظیری بر کردارها آگاهی یافت و پوشیده داشت و از نهان امور مطّلع گردید و بدان آگاه است و اقتدار و چیرگی دارد. نافرمانی گشت و آمرزید، طاعت و بندگی اش نمودند و او سپاسگزاری نمود.

فرمان روایی کرد و عدالت گسترد؛ و برتر از شائبه ی هر نقص و عیبی است و (آنچه شایسته ی هر چیزی بود، به او) عطا فرمود. همیشه بوده و هست و هیچ گاه زوال نمی یابد و چیزی همانندش نیست. و او پیش از هر چیزی است و پس از هر چیزی. پروردگاری است که به عزّتش یگانه و به قدرت خویش پادشاه (و مقتدر). و به برتری شأنش پاک (و منزّه) است. و به علوّ مقامش (به حق) خود را بزرگ می شمارد. دیده ای او را نمی بیند و نگرشی (در معرفت) بر او احاطه پیدا نمی کند. قوی و مقتدر و بینا و شنوا و برتر و حکیم و رئوف و مهربان و عزّتمند و داناست. هر آن که به توصیف او برآید، در وصفش حیران ماند. (به آفریدگان) نزدیک است و (در رفعت مقام، از آنان) دور است. .

خطبه بدون نقطه حضرت علی (ع) در مدح و ستایش خداوند

تقدیم به مهربان فرشتگانی که:

لحظات ناب باور بودن، لذت و غرور دانستن، جسارت خواستن، عظمت رسیدن و تمام تجربه های یکتا و زیبای زندگیم، مدیون حضور سبز آنهاست

تقدیم به پدر و مادر عزیزم.

بنام خدا

و من لَمْ یَشْکُر الْمَخلوق لَمْ یَشْکُر الْخالِقْ.

سپاس بی‌کران پروردگار یکتا را که هستی‌مان بخشید و به طریق علم و دانش رهنمونمان شد و به همنشینی رهروان علم و دانش مفتخرمان نمود و خوشه‌چینی از علم و معرفت را روزیمان ساخت. امیدوارم بتوانم آموخته‌هایم را در راه پیشرفت علمی وطن خویش مورد استفاده قرار دهم.
بر خود می‌دانم بدینوسیله از زحمات بی دریغ، تلاش‌های بی‌وقفه و راهنمایی‌های ارزشمند استاد گرامی جناب آقای دکتر **سعیـــد صالحـــی پـــورمهـــر**، در راستای انجام این پایان‌نامه صمیمانه تشکر و قدردانی نمایم که قطعاً بدون راهنمایی‌های ارزنده‌ی ایشان این مجموعه به انجام نمی‌رسید.
همچنین از جناب آقای دکتر **محمّـــد شهـــریـــاری** که زحمات مطالعه و مشاوره‌ی این گردایه را بر عهده داشته و در آماده سازی این رساله به نحو احسن اینجانب را مورد راهنمایی قرار دادند، کمال تشکر و قدردانی را دارم.
از جناب آقای دکتر **جعفـــرصـــادق عیـــوضلـــو** که داوری این رساله را با نهایت دقت و صرف وقت زیاد انجام دادند، تشکر می‌نمایم. از اساتید گرامی بخصوص دکتر **مرتضـــی فغفـــوری** و همچنین دکتر **اصغـــر رنجبـــری** مدیرگروه ریاضی محض که در مدت تحصیلات دانشگاهی اینجانب زحمات فراوانی را متحمل شده‌اند، تشکر می‌نمایم. در پایان از زحمات و حمایتهای بی دریغ پدر و مادرم که همیشه پشتیبانم بودند و همسر مهربانم که همراه همیشگی و پشتوانه ی زندگیم است، خواهر و برادر عزیزم که نشانه لطف الهی در زندگی من هستند و آقای علی سبزفروش اقدم کمال تشکر و قدردانی را دارم و امیدوارم جواب‌گوی این همه محبت آنها باشم و قادر به درک زیبایی‌های وجودشان باشم.

شکوفه صادقی بی‌غم
۱۳۹۳

نام خانوادگی دانشجو: صادقی بی‌غم نام: شکوفه

عنوان: تصمیم‌ناپذیری منطق مرتبه اول

استاد راهنما : سعید صالحی پورمهر

استاد مشاور : محمد شهریاری

مقطع تحصیلی: کارشناسی ارشد رشته: ریاضی محض گرایش: منطق دانشگاه تبریز

دانشکده علوم ریاضی تاریخ فارغ التحصیلی: ۱۳۹۳ تعداد صفحات: ۴۱

کلید واژه‌ها: تصمیم‌ناپذیری، نیم-تصمیم‌پذیری، ناتمامیت.

چکیده

توسط محاسبات نمادین اثبات‌های ساده‌ای برای تصمیم‌ناپذیری منطق مرتبه اول و نظریه‌های ساختارهای پایه‌ای (مانند الحاق یا حساب) به‌دست می‌آیند. با استفاده از دستور زبان‌ها یک اثبات برای نشان دادن اینکه اعتبار در منطق مرتبه اول تصمیم‌ناپذیر است (برای فرمول‌های با سور پیشوند $\exists\exists$ در زبان شامل حداقل یک رابطه یک‌تایی و یک تابع دوتایی) ارایه می‌کنیم. یک اثبات مشابه، قضیه ناتمامیت اول گودل برای ساختار رشته‌ها را نتیجه می‌دهد. تصمیم‌ناپذیری نظریه حساب با توجه به این موضوع که «اعمال جمع و ضرب می‌توانند به طور مستقیم الحاق رشته‌ها را رمز نگاری کنند» به‌دست می‌آید.

فهرست مطالب

مقدمه

توسط محاسبات نمادین، اثبات‌های ساده‌ای برای تصمیم‌ناپذیری منطق مرتبه اول و تصمیم‌ناپذیری نظریه‌های ساختارهای پایه‌ای مانند الحاق یا حساب بدست می‌آیند. کاری که ما در این پایان‌نامه (که برمبنای مقالات [۵] و [۶] نگاشته شده است) انجام می‌دهیم این است که با استفاده از دستور زبان‌ها تصمیم‌ناپذیری منطق مرتبه اول را ثابت می‌کنیم و یك اثبات برای نشان دادن اینکه اعتبار در منطق مرتبه اول تصمیم‌ناپذیر است می‌آوریم. در فصل نخست پایان‌نامه، ابتدا به بیان مفاهیم پایه در مورد دستور زبان‌ها و ماشین تورینگ می‌پردازیم. در فصل دوم تصمیم‌ناپذیری فرمول‌های وجودی را ابتدا با یك تابع دو موضعی، سپس با یك رابطه دو موضعی بیان می‌کنیم و در ادامه به بررسی نیم‌دستور زبان‌ها می‌پردازیم. در فصل سوم تصمیم‌ناپذیری ساختارهای الحاقی را ثابت می‌کنیم و در فصل آخر الحاق را به عنوان یک مبانی برای حساب معرفی می‌کنیم.

فصل ۱

تعاریف و مفاهیم اولیه

۱.۱ تعاریف اولیه

تعریف : فرض کنیم A یک مجموعه متناهی و غیر تهی باشد. اعضای A را الفبا می‌نامیم.

تعریف : هر عنصر $a \in A$ از مجموعه الفبا را حرف می‌نامیم.

تعریف : هر دنباله مرتبی از حروف یک الفبا مانند $a_1, \cdots, a_n$ را یک کلمه یا رشته می‌نامیم (تمامی عضوهای a_i باید از الفبای A باشند).

تعریف : فرض می‌کنیم A^* مجموعه همه رشته‌های الفبای A به همراه رشته تهی باشد. اگر $\mathcal{L} \subseteq A^*$، آن‌گاه $\mathcal{L}$ را یک زبان با الفبای A می‌نامیم.

برای رابطه R داریم:

$$R^2 = RoR = \{(x,z) \mid \exists y : (x,y),(y,z) \in R\}$$

$$\vdots$$

$$R_{m+1} = R^m oR$$

$$\vdots$$

$$R^+ = \bigcup_{n=1}^{\infty} R^n$$

$$R^* = \bigcup_{n=0}^{\infty} R^n$$

که $R^0 = I$.

دستور زبان : یک چهارتایی به صورت $\mathcal{G} = (\Sigma, N, S, P)$ است که در آن:

۱. Σ (مجموعه عناصر پایانی) و N (مجموعه عناصر غیرپایانی) متناهی ومجزا از هم هستند، یعنی

$\Sigma \cap N = \varnothing$.

۲. $S \in N$ که آن را نماد شروع می‌نامیم.

۳. P مجموعه متناهی از قوانین تولید $\mathcal{G}$ روی $A = \Sigma \cup N$ است، یعنی: $P \subseteq (A^*)^2$.

برای نشان دادن اینکه (u, v) قانونی از دستور زبان $\mathcal{G}$ است، می‌نویسیم: $u \longmapsto v$ (یا به صورت دقیق‌تر $u \longmapsto_{\mathcal{G}} v$). رابطه دودویی $\Rightarrow_{\mathcal{G}}$ روی A^* به صورت زیر تولید می‌شود:

اگر $u \longmapsto_{\mathcal{G}} v$ و $x, y \in A^*$ آن‌گاه $x \cdot u \cdot y \Rightarrow_{\mathcal{G}} x \cdot v \cdot y$.

اگر $S \Rightarrow_{\mathcal{G}}^* w$ آن‌گاه رشته $w \in A^*$ توسط دستور زبان $\mathcal{G}$ تولید شده است. زبان تولید شده توسط $\mathcal{G}$ به صورت زیر تعریف می‌شود:

$$\mathcal{L}(\mathcal{G}) = \{w \in \Sigma^* \mid S \Rightarrow_{\mathcal{G}}^* w\}.$$

مثال ۱.۱.۱. دستور زبان $\mathcal{G} = (\{0, 1\}, \{S, A\}, S, P)$ را با قوانین زیر در نظر بگیرید.

$$P = \begin{cases} S \longmapsto 0S \\ S \longmapsto A \\ A \longmapsto 1A \\ A \longmapsto 1 \end{cases}$$

می‌خواهیم زبان تولید شده توسط این دستور زبان را به‌دست بیاوریم. اگر درخت اشتقاق دستور زبان را با استفاده از قوانین بالا رسم کنیم خواهیم دید زبان آن به صورت زیر است که در آن m و n مستقل از هم هستند: $\mathcal{L}(\mathcal{G}) = \{0^m 1^n \mid m \geqslant 0,\ n \geqslant 1\}$. $\triangle$

ماشین تورینگ : یک شش‌تایی به صورت $T = (\mathrm{Q}, F, A, I, \tau, q_0)$ است که:

۱. Q یک مجموعه متناهی است که اعضای آن را حالت می‌نامیم.

۲. F مجموعه حالت‌های پایانی است ($F \subseteq \mathrm{Q}$).

۳. A مجموعه متناهی با یک عضو مشخص $\mathtt{B}$ (نماد خالی) است.

۴. I الفبای ورودی است ($I \subseteq A \setminus \{\mathtt{B}\}$).

۵. τ یک رابطه گذار است ($\tau \subseteq \mathrm{Q} \times A \times \mathrm{Q} \times A \times \{L, R\}$) که $\{L, R\}$ یک مجموعه دو عضوی است.

۶. $q_0 \in \mathrm{Q}$ حالت شروع است.

پیکربندی : یک پیکربندی از T یک چهارتایی (q, a, α, β) است که $q \in \mathrm{Q}$ و (a, α, β) توصیف نوار است، یعنی سرک روی حرف a بوده و α محتویات سمت راست نوار و β محتویات سمت چپ نوار (از چپ به راست) می‌باشد. بعلاوه یک پیکربندی بعضی وقت‌ها به شکل (q, w) نوشته می‌شود که $q \in \mathrm{Q}$ و w یک کلمه در A^+ با یک حرفی که زیرش خط کشیده شده است، می‌باشد.

پیکربندی c'، از پیکربندی c فقط با یک حرکت به‌دست می‌آید، اگر یکی از موارد زیر برقرار باشد:

۱. $c = (q, a, \alpha, \beta)$، $qaq'a'L \in \tau$ و $c' = (q', \beta(0), \alpha', \beta')$ که $\alpha'(0) = a'$، $\alpha'(n) = \alpha(n-1)$ برای $n > 0$ و $\beta'(n) = \beta(n+1)$ برای $n \geq 0$.

۲. $c = (q, a, \alpha, \beta)$، $qaq'a'R \in \tau$ و $c' = (q', \alpha(0), \alpha', \beta')$ که $\alpha'(n) = \alpha(n+1)$ برای $n \geq 0$، $\beta'(0) = a'$ و $\beta'(n) = \beta(n-1)$ برای $n > 0$.

یک محاسبه در T که از c شروع و در c' خاتمه می‌یابد یک دنباله‌ی متناهی

$$c = c_1, \cdots, c_n = c'$$

از پیکربندی‌هاست که $n \geq 1$ و c_{i+1} از c_i برای $1 \leq i < n$ فقط با یک حرکت به دست می‌آید. اگر c' آخرین پیکربندی به شکل (q, a, α, β) که هیچ عضوی از τ با qa شروع نمی‌شود باشد آن‌گاه محاسبه متوقف می‌شود و اگر یک محاسبه در T که از c شروع و در c' پایان می‌یابد وجود داشته باشد آن را به شکل $c \longrightarrow c'$ نشان می‌دهیم. برای $w = a_1 \cdots a_n \in A^*$ فرض کنید $c_w = (q_0, \underline{a_1} \cdots a_n)$ ($c_w = (q_0, \underline{\mathrm{B}})$ اگر $w = \varepsilon$) باشد. ماشین تورینگ T رشته w را می‌پذیرد اگر پیکربندی $c' = (q, a, \alpha, \beta)$، موجود باشد که $c_w \longrightarrow c'$ و $q \in F$ است، در این صورت زبان تولید شده توسط ماشین T به این شکل تعریف می‌شود:

$$\mathcal{L}(T) = \{w \in I^* | \text{رشته } w \text{ توسط ماشین } T \text{ پذیرفته می‌شود}\}.$$

۲.۱ حکم‌ها و ویژگی‌های ضروری

قرارداد ۱.۲.۱. فرض کنید T یک ماشین تورینگ باشد و $\mathcal{L} = \mathcal{L}(T)$ و T' همان T با حذف همه گذارهایی که با qa شروع می‌شوند و در آن $q \in F$ (F مجموعه حالت‌های پایانی) باشد. آن‌گاه

$\mathcal{L} = \mathcal{L}(T')$. واضح است که $\mathcal{L}(T') \subseteq \mathcal{L}(T)$. حال اگر $c = c_1, \cdots, c_n$ پیکربندی محاسبه ماشین تورینگ T باشد که c_n با یک حالت پایانی خاتمه می‌یابد، با در نظر گرفتن کوچکترین n ممکن، محاسبه‌ای در T' با شروع از c به‌دست می‌آید. بنابراین $\mathcal{L}(T) \subseteq \mathcal{L}(T')$.

تعریف : مساله توقف، تعیین الگوریتمی است که برای (توصیف) یک ماشین تورینگ داده شده M و یک ورودی w تعیین کند که آیا M با ورودی w بالاخره متوقف می‌شود یا نه.

قضیه ۲.۲.۱. *مساله توقف برای ماشین‌های تورینگ تصمیم‌ناپذیر است.*

برهان. فرض می‌کنیم (فرض خلف) ماشین تورینگ M' وجود دارد که مساله توقف را حل می‌کند. یک رشته توسط M' پذیرفته می‌شود اگر ورودی شامل نمایش $R(M)$ یک ماشین تورینگ M به همراه یک رشته w باشد و محاسبه M با ورودی w متوقف شود. اگر یکی از این شرایط برقرار نباشد، M' ورودی $(R(M), w)$ را نمی‌پذیرد. روش کار ماشین بدین شکل است:

$$R(M) \xrightarrow{w} \boxed{M' \text{ ماشین توقف}} \longrightarrow \begin{cases} \text{بله} \longrightarrow \text{ اگر } M \text{ با ورودی } w \text{ متوقف شود.} \\ \text{خیر} \longrightarrow \text{ اگر } M \text{ با ورودی } w \text{ متوقف نشود.} \end{cases}$$

زوج $(R(M), w)$ را به‌عنوان ورودی به ماشین توقف M' می‌دهیم. اگر M با ورودی w متوقف شود، ماشین توقف جواب بله را می‌دهد؛ و اگر M با ورودی w متوقف نشود ماشین توقف M' جواب خیر را می‌دهد. حال ماشین تورینگ M'' را بدین گونه در نظر می‌گیریم: محاسبات M'' مشابه محاسبات M' است بجز اینکه M'' برای حالتی که M' روی ورودی $(R(M), w)$ جواب بله می‌دهد، به دور نامتناهی می‌رود. تابع گذر M'' از تابع گذر M' ایجاد می‌شود. ماشین تورینگ دیگری که از ترکیب ماشین تورینگ M'' با یک ماشین D بدست می‌آید را در نظر می‌گیریم. ورودی D یک ماشین تورینگ با نمایش $R(M)$ است. یک محاسبه از D با تولید رشته $(R(M), R(M))$ از ورودی $R(M)$ شروع می‌شود. محاسبه با اجرای M'' روی $(R(M), R(M))$ ادامه می‌یابد. ورودی ماشین D ممکن است نمایش هر ماشین تورینگ با الفبای $\{0, 1, \mathtt{B}\}$ باشد. با بررسی این ماشین مشاهده می‌کنیم که D با ورودی $R(D)$ متوقف می‌شود اگر و تنها اگر D با ورودی $R(D)$ متوقف نشود. این تناقض است پس مساله توقف تصمیم‌ناپذیر است. $\square$

نتیجه ۳.۲.۱. *$\mathcal{L}_{M'}$ روی $\{0,1\}$ بازگشتی نیست. که در آن،*

$$\mathcal{L}_{M'} = \{\big(R(M), w\big) \mid \text{ماشین تورینگ } M \text{ با ورودی } w \text{ متوقف می‌شود}\}.$$

قضیه ۴.۲.۱. *اگر زبانی صوری توسط ماشین تورینگی پذیرفته شود، آن‌گاه توسط دستور زبانی تولید می‌شود.*

برهان. فرض کنید زبان $\mathcal{L}$ توسط ماشین تورینگ $T = (\mathrm{Q}, F, A, I, \tau, q_0)$ پذیرفته شود. با توجه به قرارداد ۱.۲.۱، فرض می‌کنیم که اگر $q \in F$، هیچ یک از اعضای τ با qa شروع نمی‌شوند. دستور زبان $\mathcal{G} = (\Sigma, N, S, P)$ را با $N = I$ و $\Sigma = ((I \cup \{\varepsilon\}) \times A) \cup \mathcal{Q} \cup \{S, E_1, E_2, E_3\}$ آنجایی که S, E_1, E_2, E_3 حروف اضافی هستند و P شامل موارد زیر است، در نظر می‌گیریم.

۱. $S \longrightarrow E_1E_2$.

۲. $E_2 \longrightarrow (a,a)E_2$ برای هر $a \in I$.

۳. $E_2 \longrightarrow E_3$.

۴. $E_3 \longrightarrow (\varepsilon, \mathtt{B})E_3, E_1 \longrightarrow (\varepsilon, \mathtt{B})E_1$ ($\mathtt{B}$ نماد خالی در T است).

۵. $E_3 \longrightarrow \varepsilon, E_1 \longrightarrow q_0$.

۶. $q(a, C) \longrightarrow (a, D)p$ برای هر $qCpDR \in \tau$ و $a \in I \cup \{\varepsilon\}$.

۷. $(a, C)q \longrightarrow p(a, D)$ برای هر $qCpDL \in \tau$ و $a \in I \cup \{\varepsilon\}$.

۸. $(a, C)q \longrightarrow qaq, q(a, C) \longrightarrow qaq, q \longrightarrow \varepsilon$ برای هر $a \in I \cup \{\varepsilon\}$ و $C \in A$ و $q \in F$.

نشان می‌دهیم که $\mathcal{L} = \mathcal{L}(\mathcal{G})$. فرض کنید $a_1 \cdots a_n \in I^*$ باشد. با استفاده از قوانین بالا داریم:

$$
\begin{aligned}
S &\longrightarrow E_1E_2 \\
&\longrightarrow E_1(a_1,a_1)E_2 \\
&\longrightarrow E_1(a_1,a_1)(a_2,a_2)E_2 \\
&\longrightarrow E_1(a_1,a_1)\cdots(a_n,a_n)E_2 \\
&\longrightarrow E_1(a_1,a_1)\cdots(a_n,a_n)E_3 \\
&\longrightarrow (\varepsilon,\mathtt{B})E_1(a_1,a_1)\cdots(a_n,a_n)(\varepsilon,\mathtt{B})E_3 \\
&\longrightarrow (\varepsilon,\mathtt{B})(\varepsilon,\mathtt{B})E_1(a_1,a_1)\cdots(a_n,a_n)(\varepsilon,\mathtt{B})(\varepsilon,\mathtt{B})E_3 \\
&\longrightarrow (\varepsilon,\mathtt{B})^lE_1(a_1,a_1)\cdots(a_n,a_n)(\varepsilon,\mathtt{B})^mE_3 \\
&\longrightarrow (\varepsilon,\mathtt{B})^lq_0(a_1,a_1)\cdots(a_n,a_n)(\varepsilon,\mathtt{B})^m
\end{aligned}
$$

فرض کنید $a_1 \cdots a_n$ توسط ماشین تورینگ T پذیرفته می‌شود. محاسبات با یک پیکربندی به صورت $c_0 = (q_0, \underline{a_1} \cdots a_n)$ شروع و با یک حالت در F به پایان می‌رسد. در ماشین تورینگ T، از تعداد متناهی سلول به طول l در سمت چپ رشته و تعداد متناهی سلول به طول m در سمت راست آن رشته و تعدادی سلول حاوی $a_1, \cdots, a_n$ استفاده شده؛ پس در کل از $l+m+n$ سلول استفاده شده است. پیکربندی c در محاسبات می‌تواند به شکل $c = (q, x_1 \cdots \underline{x_i} \cdots x_{l+m+n})$ شرح داده شود که $x_1, \cdots, x_{l+m+n}$ حروفی واقع روی سلول‌ها هستند. کلمه

$$.\tilde{c} = (b_1,x_1)\cdots(b_{i-1},x_{i-1})q(b_i,x_i)\cdots(b_{m+n+l},x_{m+n+l})$$

را به c نسبت می‌دهیم که در آن

$b_1 = \cdots = b_l = \varepsilon$ و $b_{l+1} = a_1, \cdots, b_{l+n} = a_n$ و $b_{l+n+1} = \cdots = b_{l+n+m} = \varepsilon$.

با استقرا روی تعداد حرکات محاسبه از c_0 تا c و با ۶ و ۷ می‌توان نشان داد که:

$$.\tilde{c}_0 = (\varepsilon,\mathtt{B})^lq_0(a_1,a_1)\cdots(a_n,a_n)(\varepsilon,\mathtt{B})^m \longrightarrow \tilde{c}$$

بنابراین: $S \longrightarrow \tilde{c}$. اگر c آخرین پیکربندی در محاسبات باشد، آن‌گاه $q \in F$ و با استفاده از ۸ داریم $a_1 \cdots a_n \longrightarrow \tilde{c}$. بنابراین $S \longrightarrow a_1 \cdots a_n$، پس $\mathcal{L}(T) \subseteq \mathcal{L}(\mathcal{G})$.

برعکس: فرض کنید $S \longrightarrow a_1 \cdots a_n$. یک اشتقاق مربوطه باید با استخراج

$$(\varepsilon, \mathtt{B})^l q_0 (b_1, b_1) \cdots (b_k, b_k)(\varepsilon, \mathtt{B})^m.$$

برای k, l, m, b_i شروع شود. استفاده از ۶ و ۷ یک حرکت در T را باعث می‌شود. بنابراین محاسبه از $(q_0, \underline{b_1} \cdots b_k)$ شروع می‌شود. سرانجام کلمه مشتق شده با استفاده از ۸ باید شامل $q \in F$ باشد. بنابراین T، $b_1 \cdots b_k$ را می‌پذیرد. ادامه اشتقاق با استفاده از ۸ به‌دست می‌آید و در $b_1 \cdots b_k$ پایان می‌یابد. بنابراین

$$b_1 \cdots b_k = a_1 \cdots a_n$$

توسط T پذیرفته می‌شود، پس $\mathcal{L}(T) = \mathcal{L}(\mathcal{G})$. □

فصل ۲

تصمیم‌ناپذیری فرمول‌های وجودی

۱.۲ تصمیم‌ناپذیری فرمول‌های وجودی

چون دستور زبان‌ها تورینگ-کامل هستند، پس طبق قضیه ۴.۲.۱ دستور زبان $\mathcal{G}$ در الفبای A وجود دارد به طوری که $\mathcal{L}(\mathcal{G})$ تصمیم‌ناپذیر است. فرض کنید $V_{\mathcal{G}}$ زبانی مرتبه اول باشد که شامل موارد زیر است:

۱. نمادهای ثابتی ε و نیز $\overline{\alpha}$ برای هر $\alpha \in A$.

۲. یک نماد رابطه‌ایی تک موضعی $\mathtt{G}$.

۳. یک نماد تابعی دو موضعی مانند $*$. برای رشته $w = \alpha_1 \cdots \alpha_k \in A^*$، رشته $\overline{w}$ را برابر $V_{\mathcal{G}}$-ترم به صورت $\alpha_1 * (\alpha_2 * (\cdots (\alpha_{k-1} * \alpha_k) \cdots))$ می‌گیریم. فرض کنید $\varphi_{\mathcal{G}}$ یک $V_{\mathcal{G}}$-فرمول است که توسط ترکیب عطفی فرمول‌های φ_1، φ_2 و φ_3 تعریف شده است:

(φ_1) $\mathtt{G}(\varepsilon)$.

(φ_2) برای هر قانون $u \longmapsto_{\mathcal{G}} v$، $\forall x, y \left[\mathtt{G}(x * (\overline{u} * y)) \longrightarrow \mathtt{G}(x * (\overline{v} * y))\right]$.

(φ_3) بستار $\mathtt{G}$ تحت شرکت‌پذیری $*$. این می‌تواند فقط با دو سور به صورت ترکیب عطفی برای $\alpha \in A$ از فرمول‌ها بیان می‌شود:

$$\forall x \forall y \left[\mathtt{G}(x * (\alpha * y)) \longleftrightarrow \mathtt{G}((x * \alpha) * y)\right].$$

لم ۱.۱.۲. *برای هر رشته $w \in A^*$، اگر $S \Rightarrow^*_{\mathcal{G}} w$ آنگاه*

$$\models \varphi_{\mathcal{G}} \longrightarrow \mathtt{G}(\overline{w}) . \qquad (۱.۲)$$

برهان. با استقرا روی n ثابت می‌کنیم که اگر $S \Rightarrow^n_{\mathcal{G}} w$ آن‌گاه رابطه (۱.۲) برقرار است. برای $n = 0$ از (φ_1) نتیجه می‌شود. ادعا را برای n فرض می‌کنیم. اگر

$$S \Rightarrow^n_{\mathcal{G}} x \cdot u \cdot y \Rightarrow x \cdot v \cdot y$$

با قانون $v \longmapsto u$ به دست آمده باشد، آن گاه با فرض استقراء داریم $\models \varphi_{\mathcal{G}} \longrightarrow \mathtt{G}(\overline{x \cdot u \cdot y})$
وهمین‌طور از (φ_3)، $\models \varphi_{\mathcal{G}} \longrightarrow \mathtt{G}(\overline{x} * \overline{u} * \overline{y})$ که با (φ_2) نتیجه می‌دهد $\models \varphi_{\mathcal{G}} \longrightarrow \mathtt{G}(\overline{x} * \overline{v} * \overline{y})$
دوباره از (φ_3) داریم $\models \varphi_{\mathcal{G}} \longrightarrow \mathtt{G}(\overline{x \cdot v \cdot y})$. $\square$

لم ۲.۱.۲. *فرض کنید $\mathcal{S}$ یک $V_{\mathcal{G}}$-ساختار با جهان تمامی $V_{\mathcal{G}}$-ترم‌ها باشد که در آن تعبیر* $\mathtt{G}$ *برابر است با* $\{$ w *نشان داده شده توسط* t *در* $\mathcal{L}(\mathcal{G})$ *است* $|t\} = \mathtt{G}_{\mathcal{S}}$. *در این صورت داریم:*

$$\mathcal{S} \models \varphi_{\mathcal{G}}.$$

برهان. کافی است نشان دهیم که $\mathcal{S} \models \varphi_1 \wedge \varphi_2 \wedge \varphi_3$.
برای $\mathcal{S} \models \varphi_1$ توجه داریم که از $\mathcal{S} \Rightarrow^*_{\mathcal{G}} \mathcal{S}$ فوراً $\mathcal{S} \models \mathtt{G}(\varepsilon)$ نتیجه می‌شود. برای $\mathcal{S} \models \varphi_2$ اگر $\mathcal{S} \models \mathtt{G}(x * \overline{u} * y)$ آنگاه $x \cdot \overline{u} \cdot y \in \mathcal{L}(\mathcal{G})$. پس از قانون $u \longmapsto_{\mathcal{G}} v$ خواهیم داشت $x \cdot \overline{v} \cdot y \in \mathcal{L}(\mathcal{G})$.
بنابراین $\mathcal{S} \models \mathtt{G}(x * \overline{v} * y)$. بالاخره برای $\mathcal{S} \models \varphi_3$ چون $x\alpha y = (x\alpha)y$ برقرار است، پس $(x\alpha)y \in \mathcal{L}(\mathcal{G}) \longleftrightarrow x\alpha y \in \mathcal{L}(\mathcal{G})$. بنابراین $S \models \mathtt{G}(x * \alpha y) \longleftrightarrow \mathtt{G}(x\alpha y)$. $\square$

نتیجه ۳.۱.۲. *اگر* $\models \varphi_{\mathcal{G}} \longrightarrow \mathtt{G}(\overline{w})$ *آنگاه* $w \in \mathcal{L}(\mathcal{G})$.

برهان. اگر $\models \varphi_{\mathcal{G}} \longrightarrow \mathtt{G}(\overline{w})$ آنگاه برای ساختار ترمی $\mathcal{S}$ بالا $\mathcal{S} \models \varphi_{\mathcal{G}} \longrightarrow \mathtt{G}(\overline{w})$. چون $\mathcal{S} \models \varphi_{\mathcal{G}}$، داریم $\mathcal{S} \models \mathtt{G}(\overline{w})$، و در نتیجه $w \in \mathcal{L}(\mathcal{G})$. $\square$

۲.۲ تصمیم‌ناپذیری فرمول‌های وجودی با یک تابع دو موضعی

رده‌بندی فرمول‌ها را بدین شکل در نظر می‌گیریم: $[Q, (r_1, r_2), (f_1, f_2)]$ برابر است با مجموعه فرمول‌های پیشوندی با پیشوند سوری Q که در آن r_i یک رابطه‌ی i-موضعی و f_i یک تابع i-موضعی می‌باشد $(i = 1, 2)$. اگر $r_2 = 0$ باشد، آن را حذف می‌کنیم و همین‌طور f_2 را.

قضیه ۱.۲.۲. *درستی فرمول‌های* $[\exists\exists, (1), (0, 1)]$ *تصمیم‌ناپذیر است.*

برهان. فرض کنید $\mathcal{G}$ دستور زبانی باشد که $\mathcal{L}(\mathcal{G})$ تصمیم‌ناپذیر است و $\varphi_{\mathcal{G}}$ متناظر با فرمول‌های تعریف شده در بالا باشد. آنگاه برای هر $w \in \Sigma^*$،

$$[w \in \mathcal{L}(\mathcal{G})] \iff [S \Rightarrow^*_{\mathcal{G}} w].$$

بنابر لم ۱.۱.۲ و نتیجه ۳.۱.۲، $[w \in \mathcal{L}(\mathcal{G})] \iff [\ \models \varphi_{\mathcal{G}} \longrightarrow \mathtt{G}(\overline{w})]$. بنابراین $\mathcal{L}(\mathcal{G})$ به مساله درستی برای فرمول‌های $(w \in \Sigma^*)$ $\varphi_{\mathcal{G}} \longrightarrow \mathtt{G}(\overline{w})$، تحویل یافته است، که فرم‌های پیشوندی در آن $[\exists^2, (1), (0,1)]$ هستند. در نتیجه مساله اعتبار برای این رده از فرمول‌ها تصمیم‌ناپذیر است. □

۲.۳ تصمیم‌ناپذیری فرمول‌های وجودی با یک رابطه‌ی دو موضعی

اثبات فوق مبتنی بر تعریف‌پذیری مجموعه $\{w \in A^* \mid S \Rightarrow^*_{\mathcal{G}} w\}$ است. می‌توانیم به جای آن مجموعه

$$\{(z,w) \in A^2 \mid z \Rightarrow^*_{\mathcal{G}} w\}$$

را در نظر بگیریم، که بدون استفاده از الحاق شرح داده می‌شود: فرض کنید $V'_{\mathcal{G}}$ زبانی با یک نماد ثابت منحصر به فرد ε باشد و برای هر $\alpha \in A$ یک نماد تابعی یک موضعی $\overline{\alpha}$ و نیز یک نماد رابطه‌ای دو موضعی T را در نظر می‌گیریم. اگر $u = \alpha_1\alpha_2\cdots\alpha_k \in A^*$، قرار دهید $\overline{u} = \overline{\alpha_1}(\overline{\alpha_2}(\cdots\overline{\alpha_k}(\varepsilon)\cdots))$. به عبارت دیگر $\overline{u}$ یک نمایش برای u به‌صورت یک $V'_{\mathcal{G}}$- ترم است. همچنین اگر t یک $V'_{\mathcal{G}}$- ترم باشد، به‌جای $\overline{\alpha_1}(\overline{\alpha_2}(\cdots\overline{\alpha_k}(t)\cdots))$، $\overline{u}(t)$ می‌نویسیم. فرض کنید $\psi_{\mathcal{G}}$ ترکیب عطفی فرمول‌هایی باشد که بیان می‌کنند رابطه دوتایی T خواص بستاری $\Rightarrow^+_{\mathcal{G}}$ را ارضا می‌کند، یعنی خواص تراگذاری $\Rightarrow_{\mathcal{G}}$ (به پیمانه نمایش کلمات توسط ترم‌ها):

(ψ_1) برای هر $u \longmapsto_{\mathcal{G}} v$، فرمول $\forall x\, T(\overline{u}(x), \overline{v}(x))$ را داریم.

(ψ_2) برای هر $\alpha \in A$، فرمول $\forall x, y\, [T(x,y) \longrightarrow T(\overline{\alpha}(x), \overline{\alpha}(y))]$ را داریم.

(ψ_3) بستار T تحت خاصیت تراگذاری به این صورت است:

$$.\forall x, y, z\, [T(x,y) \wedge T(y,z) \longrightarrow T(x,z)]$$

مقایسه لم ۱.۱.۲ ونتیجه ۳.۱.۲ به راحتی ثابت می‌کند که برای $w, z \in A^*$: $z \Rightarrow^+_{\mathcal{G}} w$ اگر و فقط اگر $\models \psi_{\mathcal{G}} \longrightarrow T(\overline{z}, \overline{w})$ و در نتیجه $[w \in \mathcal{L}(\mathcal{G})] \iff [\ \models \psi_{\mathcal{G}} \longrightarrow T(S, \overline{w})]$. با انتخاب دستور زبان $\mathcal{G}$ که $\mathcal{L}(\mathcal{G})$ تصمیم‌ناپذیر است نتیجه می‌گیریم که درستی فرمول‌هایی به شکل $\psi_{\mathcal{G}} \longrightarrow T(S, \overline{w})$ تصمیم‌ناپذیر است. چون سور پیشوندی در $\varphi'_{\mathcal{G}}$، $\forall^3$ هستند، داریم: اگر دستور زبان $\mathcal{G}$ روی n متغیر طوری وجود داشته باشد که $\mathcal{L}(\mathcal{G})$ تصمیم‌ناپذیر است، آنگاه درستی

$[\exists^3, (0,1), (n)]$ تصمیم‌ناپذیر است. هنوز هم با وجود سادگی اثبات ما و ارزش آموزشی آن، این نتیجه کمی ضعیف‌تر از تصمیم‌ناپذیری اعتبار برای $[\exists^3, (0,1), (1)]$ ارایه شده در [۴] است.

۴.۲ موضوع جانبی: نیم دستور زبان‌ها

در دستور زبان‌ها یک استنتاج پایانی صریحا با نبودن نمادهای غیر پایانی نشان داده می‌شود. به هر حال یک استنتاج ممکن است قابل گسترش نباشد، هرگاه هیچ قانونی در رشته‌های آخری صدق نکند. ما اینجا نوع دیگری از دستور زبان‌ها را بررسی می‌کنیم.

یک نیم‌دستور زبان P روی الفبای Σ، شامل یک رشته $S \in \Sigma^*$ به عنوان نماد شروع و یک مجموعه از قوانین $u \longmapsto v$ با $u \in \Sigma^+$ و $v \in \Sigma^*$ است. اگر $u \longmapsto v$ یک قانون باشد،آنگاه برای همه $x, y \in \Sigma^*$ داریم $x \cdot u \cdot y \Rightarrow_P x \cdot v \cdot y$. اگر هیچ zی وجود نداشته باشد که $w \Rightarrow_P z$، آن را با نماد $P \downarrow w$ نشان می‌دهیم و $\mathcal{L}(P)$ را به این صورت تعریف می‌کنیم:

$$\mathcal{L}(P) = \{w \mid S \Rightarrow_P^* w \wedge P \downarrow w\}.$$

قضیه ۱.۴.۲. *نیم‌دستور زبان‌هایی مانند P روی $\{0,1\}$ وجود دارند به طوری که $\mathcal{L}(P)$ تصمیم‌ناپذیر است.*

برهان. فرض کنید $\mathcal{G}$ یک دستور زبان روی الفبای Σ با متغیرهای N $A = \Sigma \cup N$ باشد به طوری که $\mathcal{L}(\mathcal{G})$ تصمیم‌ناپذیر است. یک کد گذاری $\beta : A \longrightarrow \{0,1\}^*$ انتخاب می‌کنیم به طوری که برای هر $\alpha \in A$، $\beta(\alpha)$ به فرم $00\alpha_1 0\alpha_2 0 \cdots 0\alpha_r 0111$ است و هر α_i 1 یا 11 می‌باشد. برای $w = \alpha_1 \cdots \alpha_r \in A^*$، $\beta(w)$ را به صورت $\beta(w) = \beta(\alpha_1) \cdots \beta(\alpha_r)$ تعریف می‌کنیم. توجه کنید که تجزیه $\beta(\alpha_1 \cdots \alpha_r)$ به $\beta(\alpha_1), \cdots, \beta(\alpha_r)$ با انتخاب β به طور یکتا مشخص می‌شود. بنابراین اگر z یک زیر کلمه از $\beta(w)$ باشد،آنگاه

$(*)$ برای هر xی، $z = \beta(x)$ اگر و فقط اگر z به فرم $00z_0111$ باشد و در این حالت x یک زیر کلمه از w است.

حال دستور زبان $\mathcal{G}$ را با کدگذاری β به یک نیم‌دستور زبان P^β روی $\{0,1\}$ تبدیل می‌کنیم: رشته

آغازین $\beta(S)$ بوده و قوانین به صورت زوج مرتب $(\beta(u), \beta(v))$ هستند که (u, v) قانونی از $\mathcal{G}$ می‌باشد. چون $(*)$ برقرار است داریم: $[w \Rightarrow_{\mathcal{G}} z] \iff [\beta(w) \Rightarrow_P \beta(z)]$. $\square$

حال نتیجه اصلی‌مان را به صورت زیر تظریف می‌کنیم:

قضیه ۲.۴.۲. *مساله اعتبار برای فرمول‌های کلاس* $[\exists^3, (0,1), (2)]$ *تصمیم‌ناپذیر است.*

فصل ۳

ناتمامیت گودل و ساختارهای الحاقی تصمیم‌ناپذیر

۱.۳ رشته‌ها با الحاق

برای الفبای $\Xi = \{\xi_1, \cdots, \xi_k\}$ فرض کنید W_Ξ زبانی با نمادهای ثابت $\varepsilon, \xi_1, \cdots, \xi_k$ و یک نماد تابعی دو موضعی که آن را به صورت میان‌وندی $*$ نشان می‌دهیم، باشد. با این پیش فرض که نمادهای سمت راست $*$ در الویت قرار می‌گیرند. فرض کنید $\mathcal{W}_\Xi = (\Xi^*; \xi_1, \cdots, \xi_m; *)$، یک W_Ξ-ساختار باشد، که $*$ تابع الحاق می‌باشد.

قضیه ۱.۱.۳. *الفبای Ξ چنان وجود دارد که نظریه ساختار $\mathcal{W}_\Xi$، $Th(\mathcal{W}_\Xi)$، نیم-تصمیم‌پذیر نیست.*

برهان. فرض کنید $\mathcal{G}$ دستور زبانی با الفبای $A = \Sigma \cup N = \{\alpha_1, \cdots, \alpha_k\}$ باشد که $\mathcal{L}(\mathcal{G})$ تصمیم‌ناپذیر است و $\Xi = A \cup \{\$\}$ که $\$ \notin A$. (از $\$$ برای نمایش ساختار ترتیبی $(w_1 \cdots w_k)$ روی A با یک کلمه یکتای $\$w_1\$ \cdots \$w_k\$ \in \Xi^*$ استفاده خواهیم کرد). برای تعریف $\mathcal{L}(\mathcal{G})$ با یک W_Ξ -فرمول، فرمول‌های زیر را در نظر می‌گیریم.

$$Tr_{uv}[w,z] \equiv w \Rightarrow z \; [\text{با بکار بردن قانون } u \mapsto v]$$

$$\equiv \exists p, q \left(w = p * u * q \ \wedge z = p * v * q\right)$$

$$Tr_{\mathcal{G}}[w,z] \equiv w \Rightarrow_{\mathcal{G}} z$$

$$\equiv \bigvee \{Tr_{uv}(w,z) \mid \text{قانونی از } \mathcal{G} \text{ است } u \mapsto v\}$$

$$w \sqsubseteq z \equiv \exists x, y \; (z = x * w * y)$$

$$Drv_{\mathcal{G}}[c] \equiv \text{بخش } c \text{ بین دو نماد } \$ \text{ یک } \mathcal{G} - \text{اشتقاق است}$$

$$\equiv \forall w, z \left(\$ * w * \$ * z * \$ \sqsubseteq c \ \wedge \ \neg(\$ \sqsubseteq w * z) \longrightarrow Tr_{\mathcal{G}}[w,z]\right)$$

$$In_{\mathcal{G}}[w] \equiv \text{یک } \mathcal{G} - \text{اشتقاق از } w \text{ وجود دارد}$$

$$\equiv \exists c \; Drv_{\mathcal{G}}[\$ * \varepsilon * \$ * c * \$ * w * \$]$$

به وضوح $w \notin \mathcal{L}(\mathcal{G})$ اگر و فقط اگر $\mathcal{W}_\Xi \models \neg In_{\mathcal{G}}[\overline{w}]$. اما $\mathcal{L}(\mathcal{G})$ تصمیم‌ناپذیر است، پس $\Xi^* - \mathcal{L}(\mathcal{G})$ نیم-تصمیم‌پذیر نیست. بنابراین درستی فرمول‌هایی به شکل $\neg In_{\mathcal{G}}[\overline{w}]$ در $\mathcal{W}_\Xi$ نیم-تصمیم‌پذیر نیست. □

نتیجه ۲.۱.۳. *هیچ نظریه نیم-تصمیم‌پذیر که بستار استنتاجی آن $Th(\mathcal{W}_\Xi)$ باشد، وجود ندارد.*

برهان. بستار استنتاجی هر نظریه نیم-تصمیم‌پذیر، نیم-تصمیم‌پذیر است [۳]. □

با استفاده از کدگذاری β از بخش ۴.۲ فصل دوم، اثبات‌های بالا را می‌توان به دو حرف از الفبای Ξ تعدیل کرد. بنابراین قضیه زیر را داریم:

قضیه ۳.۱.۳. *$Th(\mathcal{W}_{\{0,1\}})$ نیم-تصمیم‌پذیر نیست.*

۲.۳ محاسبه با جمع و ضرب

ساختار $\mathcal{N} = (\mathbb{N}; 0, 1; +, \times)$ را با جمع و ضرب اعداد طبیعی در نظر می‌گیریم. نشان می‌دهیم که $Th(N)$ با تقلیل $Th(\mathcal{W}_{\{0,1\}})$ به $Th(\mathcal{N})$ نیم-تصمیم‌پذیر نیست. بعضی از اثبات‌ها برای $\mathcal{W}_\Xi$، برای هر الفبای Ξ با طول بزرگتر از ۱ کار می‌کنند. برای رشته $w \in \{0,1\}^*$، فرض می‌کنیم $v(w)$ مقدار عدد دودویی $1w$ باشد. به عنوان مثال ε با عدد ۱ و رشته 001 با ۹ نشان داده می‌شود. فرمول $Ct[a,b,c]$ را در زبان $\mathcal{N}$ طوری می‌سازیم که $\mathcal{N} \models Ct[v(w), v(z), v(x)]$ اگر و تنها اگر $x = w*z$. نخست اختصارات زیر را معرفی می‌کنیم:

$$\begin{aligned}
n \le m \;&\equiv\; \exists d\; (n + d = m)\\
n < m \;&\equiv\; n \le m \;\wedge\; n \neq m\\
n \mid m \;&\equiv\; \exists q\; (n \times q = m)\\
P[y] \;&\equiv\; \text{«}y \text{ توانی از } 2 \text{ است»}\\
&\equiv\; \forall m\; (1 < m \;\wedge\; m \mid y \longrightarrow 2 \mid m)\\
L[y,m] \;&\equiv\; \text{«}y \text{ توانی از } 2 \text{ است و کلمات ارایه شده توسط } y \text{ و } m \text{ طول یکسانی دارند»}\\
&\equiv\; P[y] \;\wedge\; y \le m \;\wedge\; m < 2 \cdot y\\
Ct[n,m,p] \;&\equiv\; \exists y\; \Big(L[y,m] \;\wedge\; p + y = n \cdot y + m\Big)
\end{aligned}$$

برای مثال فرض کنید $w = 001$ و $z = 1100$ باشند. پس $v(w) = [1001]_2 = 9$ و $v(z) = [11100]_2 = 28$. توجه داشته باشید که $L[y, v(z)]$ اگر و فقط اگر $y = [10000]_2 = 16$ و $Ct[v(w), v(z), c]$ اگر و فقط اگر $c = v(w) \cdot y + v(z) - y$ به بیان دیگر در عالم دودویی

$$c = 1001.1000 + (11100 - 10000) = 1001000 + 1100 = 10011100.$$

اما $v(w * z) = v(0011100) = 100111010$، به بیان دیگر $Ct(v(w), v(z), c)$ اگر و فقط اگر $c = v(w * z)$.

در حالت کلی داریم:

- y توانی از ۲ است اگر و فقط اگر هر مقسوم‌علیه آن زوج داشته باشد. پس y توانی از ۲ است اگر و فقط اگر $P[y]$.
- اگر m در حساب دودویی k رقم داشته باشد آن‌گاه $y = 2^k$ اگر و فقط اگر y توانی از ۲ باشد و $y \leqslant m < 2y$. به عبارت دیگر اگر و فقط اگر $L[y, m]$.
- فرض کنید اعداد دودویی برای n و m به ترتیب $1w$ و $1z$ باشد. اگر y طوری باشد که $L[y, m]$، آنگاه نمایش دودویی برای y برابر 10^k است که k طول m است. بنابراین $1wz$ عدد دودویی برای $ny + m - y$ است. پس لم زیر اثبات شده است.

لم ۱.۲.۳. *برای هر* $w, z, x \in \{0, 1\}^*$، $\mathcal{N} \models Ct[v(w), v(z), v(x)]$ *اگر و فقط اگر* $x = w * z$.

قضیه ۲.۲.۳. $Th(\mathcal{N})$ *نیم-تصمیم‌پذیر نیست.*

برهان. اثبات قضیه ۳.۱.۳ به طوری دیگر با استفاده از کدهای عددی کلمه‌ها به جای خود کلمه‌ها، با رابطه عددی Ct به عنوان الحاق، نتیجه مورد نظر را بدست می‌دهد.

روش دیگر اثبات قضیه استفاده از لم ۱.۲.۳ است که ساختار $\mathcal{W}_{\{0,1\}}$ را در $\mathcal{N}$ تعریف می‌کند. □

نتیجه ۳.۲.۳. (*قضیه اول ناتمامیت گودل*)

هیچ نظریه نیم-تصمیم‌پذیری وجود ندارد که بستار استنتاجی آن $Th(\mathcal{N})$ *باشد.*

فصل ۴

الحاق به عنوان یک مبانی برای حساب

۱.۴ مقدمه

در این فصل عملی به نام «الحاق» را در نظر می‌گیریم که با علامت «$*$» نشان داده شده و به این صورت درک می‌شود: هرجا x و y به شکل $x * y$ نوشته شود، x و y بدون فاصله پشت سرهم قرار می‌گیرند؛ به عنوان مثال $\alpha * \beta$، به معنی رشته $\alpha\beta$ است. سپس با استفاده از نظریه الحاق، اعمال جمع و ضرب و توان و $\cdots$ را تعریف می‌کنیم.
در این فصل ما از عمل الحاق به عنوان یک پایه برای چندین ساختار استفاده خواهیم کرد. با توجه به ساختارهایی که ما اینجا بررسی می‌کنیم می‌توان نشان داد که علم حساب اعداد طبیعی را می‌توان به جای نظریه مقدماتی الحاق جایگذاری کرد. البته اینجا از علم حساب اعدا طبیعی به صورت مقدماتی استفاده نمی‌کنیم اما اصول مقدماتی مذبور برای ما مهم است. مولفه‌های اتمی را با α و β نمایش می‌دهیم و تعداد این مولفه‌ها نمی‌تواند کمتر از ۲ باشد.

۲.۴ الحاق در علم حساب

نشان داده خواهد شد که نه تنها حساب مقدماتی اعداد طبیعی می‌تواند در نظریه مقدماتی الحاق نشانده شود بلکه این نشاندن می‌تواند چنان انجام شود که اثبات یکسان بودن نظریه مقدماتی اعداد طبیعی و نظریه مقدماتی الحاق را نتیجه دهد. اینجا روی اعداد طبیعی که همان اعداد صحیح مثبت هستند بحث می‌کنیم.
لازم نیست نظریه الحاق تعبیر معنایی داشته باشد و نباید به نظریه الحاق به عنوان یک موضوع نحوی نگاه کنیم که چیزی فراتر از آن است. اشیائی که در این نظریه با آن‌ها سروکار داریم باید رشته‌هایی متناهی دلخواهی از اشیاء باشند که اتم نامیده می‌شوند. اشیاء (یا اتم‌ها) خودشان به عنوان دنباله تلقی می‌شوند یا به‌طور مختصر دنباله‌هایی به طول یک هستند. الحاق عملی است که از پشت سر هم قرار دادن دنباله‌ها، دنباله‌ای جدید می‌سازد. بنابراین اگر x دنباله‌ای شامل اتم‌های متوالی a, b, b, a, c بوده و y دنباله شامل اتم‌های متوالی b, a (به ترتیب از چپ به راست) باشند، دنباله $x * y$ به شکل a, b, b, a, c, b, a می‌باشد. با این قرارداد که ما از سمت چپ عمل الحاق را انجام می‌دهیم، می‌توانیم

از گذاشتن پرانتزها صرف‌نظر کنیم یعنی از آن‌جایی که عمل الحاق شرکت‌پذیر است، واضح است که گروه‌بندی در این‌جا (ترتیب اعمال الحاق) فرقی نمی‌کند.
اگر تعداد اتم‌ها را ۲ فرض کرده و آن‌ها را با a و b نشان دهیم، فرمول‌های مرتبه اول نظریه مقدماتی الحاق در زبان $\{a, b, *, =\}$ است که در آن a و b نمادهای ثابتی، $*$ نماد تابعی دو موضعی و $=$ نماد رابطه‌ای دو موضعی است. نظریه مقدماتی اعداد صحیح مثبت شامل نظریه‌ای است که می‌توان آن را در قالب اعمال جمع و ضرب و توان نمایش داد. بنابراین فرمول‌های مرتبه اول حساب مقدماتی مرتبه اول در زبان $\{+, \times, \uparrow\}$ هستند که در آن $+$ (جمع)، $\times$ (ضرب) و $\uparrow$ (توان) نمادهای تابعی دوموضعی هستند.
ثابت خواهد شد که حساب را می‌توان در زبان نظریه الحاق به نحوی ارایه داد که تمام مفاهیم حساب مقدماتی از اعداد صحیح مثبت در این زبان قابل تعریف باشند. یکسان‌سازی اعداد صحیح مثبت با دنباله‌ها می‌تواند یک حرفی باشد به این معنی که اعداد صحیح همگی با زیر کلاسی از دنباله‌ها یکسان‌سازی می‌شوند و تابع نمایش اعداد که از اعداد صحیح مثبت به مجموعه دنباله‌ها ($\mathbb{N} \hookrightarrow \Sigma^*$) است، تابعی یک به یک و غیر پوشا می‌باشد. همچنین می‌تواند دو حرفی باشد به این معنی که اعداد صحیح و دنباله‌ها چنان با هم یکسان‌سازی شده‌اند که بین هر دوی آن‌ها تناظر یک‌به‌یک و پوشا وجود داشته باشد. تمام اعداد طبیعی با تمام دنباله‌های متناهی تناظر یک‌به‌یک و پوشا دارند. یعنی هر عدد طبیعی را می‌توان با یک دنباله متناهی نشان داد و هر دنباله متناهی نشان‌دهنده‌ی یک عدد طبیعی است.

۳.۴ روابط متناهی

فرض خواهیم کرد نظریه الحاق که در این فصل در مورد آن بحث می‌کنیم شامل دو اتم a و b و یا بیشتر باشد. یک رابطه متناهی ممکن است به شکل یک مجموعه متناهی از زوج‌های مرتب باشد و به صورت لیست دو ستونی نشان داده شود. حال هر رابطه‌ای به این شکل را در نظر می‌گیریم:
فرض می‌کنیم (u_1, v_1)، (u_2, v_2)، $\cdots$، (u_n, v_n) زوج‌مرتب‌هایی باشد که خود u_1، v_1، u_2 و $\cdots$ به صورت دنباله هستند. فرض کنید z یک دنباله از aها باشد به‌طوری که طولانی‌تر از هر علامت

ظاهر شده و دنباله‌های u_1، v_1، u_2 و $\cdots$ می‌باشد. حال دنباله‌های $w_1, ..., w_n$ را به صورت زیر در نظر می‌گیریم:

$$w_1 = b * z * b * u_1 * b * z * b * v_1 * b * z$$

$$w_2 = b * z * b * u_2 * b * z * b * v_2 * b * z$$

.

.

.

$$w_n = b * z * b * u_n * b * z * b * v_n * b * z$$

و همه این‌ها در یک دنباله ساده خلاصه می‌شود:

$$.w = w_1 * w_2 * \cdots * w_n$$

حال ویژگی‌های متفاوتی از w را در نظر می‌گیریم:
(i) فقط رخدادهایی از z در w هستند که این رخدادها به طور آشکار در بسط w_1، w_2، $\cdots$، w_n نشان داده شده و هیچ رخداد اضافی پنهان در z وجود ندارد و به این شکل به نظر می‌رسد که هیچ رخداد اضافی از z نمی‌تواند وجود داشته باشد زیرا هر رخدادی که به‌طور آشکار نشان داده می‌شود در ابتدا و انتهایش با اتم b مجزا شده است.
(ii) تنها بخش‌هایی از w که $z * b$ قبل و $b * z$ بعد از آن‌ها آمده و شامل هیچ رخدادی از z نمی‌شوند، شامل u_1، v_1، u_2 و $\cdots$ هستند که به‌طور آشکار در w_1، $\cdots$، w_n نشان داده شده‌اند.
(iii) اگر x و y دنباله‌هایی باشند که هیچ کدام شامل رخدادی از z نباشد و اگر

$$z * b * x * b * z * b * y * b * z \qquad (۱.۴)$$

در بخشی از w رخ بدهد، آنگاه x و y باید برای هر iی برابر با u_i و v_i باشند. چونکه بنابه (ii) x باید یکی از u_1، v_1، u_2 و $\cdots$ بوده و y باید یکی بعد از آن باشد. هنوز رخدادهایی از x و y که از بخش‌های w_1، $\cdots$، w_n موجود در w متفاوت باشند، وجود ندارند زیرا اگر وجود داشت به عنوان مثال می‌توانستیم $b * z * b$ را به جای $b * z * b * z * b$ در نظر بگیریم. بنابراین x و y باید برای هر

iی برابر با u_i و v_i باشند.

با استفاده از (iii) نتیجه زیر را داریم:

در روابط متناهی می‌توان کلمه رابطه را حذف و به جای آن دنباله w را بیاوریم و بگوییم رابطه‌ی (۱.۴) بخشی از w است و x و y شامل هیچ رخدادی از z نیستند. هرگاه بنویسیم «$w(x,y)$» بدین معنی است که (۱.۴) بخشی از w بوده و x و y شامل هیچ رخدادی از z نیستند (z طولانی‌ترین رشته‌ای از aها است که در w ظاهر می‌شود). پس می‌توانیم دنباله w را طوری بسازیم به‌طوری‌که برای هر x و y، x در طرف چپ y قرار می‌گیرد و $w(x,y)$ معادل با این جمله است: x با y رابطه دارد. به‌طور مقدماتی برای تعریف رسمی «$w(x,y)$» لازم است علامت «$\sqsubseteq$» را معرفی کنیم که به معنی «رخ دادن در»، «بخشی از» می‌باشد که اینجا همان زیرکلمه (زیر رشته) است. تعریف این مفهوم توسط نظریه الحاق به‌صورت زیر است:

$$D1. \qquad x \sqsubseteq y \equiv (\exists z)(\exists w)\big((x=y) \vee (z*x=y) \vee (x*w=y) \vee (z*x*w=y)\big)$$

پس «$x \sqsubseteq y$» به این معنی است که: دنباله x شامل بخش یا کل دنباله y است، یا به عبارتی x زیر رشته‌ای از y است. محمول بعدی $T(z)$ است و به این معنی است که: z رشته‌ای است که فقط شامل a است ($z = aaaa\cdots a$) و به شکل زیر تعریف می‌شود:

$$D2. \qquad T(z) \equiv \forall x(x \sqsubseteq z \longrightarrow a \sqsubseteq x)$$

گوییم z طولانی‌ترین رشته شامل a و مشمول در w است هرگاه z فقط شامل a بوده و هر زیر رشته w که فقط شامل a باشد زیر رشته‌ای از z است، یعنی

$$T(z) \wedge z \sqsubseteq w \wedge \forall v\big[T(v) \wedge v \sqsubseteq w \longrightarrow v \sqsubseteq z\big]$$

که با

$$\forall v\big[T(v) \wedge v \sqsubseteq w \longleftrightarrow v \sqsubseteq z\big] \qquad (۲.۴)$$

معادل است.

حال تعریف $w(x,y)$ به صورت زیر است:

«دنباله‌ی z که (۲.۴) را ارضا می‌کند و بخشی از x یا y نیست وجود دارد و دنباله‌ای که در (۱.۴) بیان شده بخشی از w است».

$$D3. \qquad w(x,y) \equiv (\exists z)\Big[\forall v\ \big(T(v) \wedge v \sqsubseteq w \longleftrightarrow v \sqsubseteq z\big) \wedge (z \not\sqsubseteq x)\ \wedge\ (z \not\sqsubseteq y)\ \wedge\ z * b * x * b * z * b * y * b * z \sqsubseteq w\Big]$$

استدلال بخش حاضر نشان می‌دهد که برای هر رابطه متناهی R از دنباله‌ها، دنباله w وجود دارد به‌طوری که:

$$\forall x\ \forall y\ [w(x,y) \equiv xRy]$$

خواهیم دید که چطور لیستی از زوج‌های مرتب شامل روابط را می‌دهد به‌طوری که دنباله w واقعا می‌تواند ساخته شود.

۴.۴ ساختار یک حرفی

روش راحتی برای بدست آوردن مدل اعداد صحیح مثبت توسط دنباله‌هایی که پایه آن‌ها ۲ یا تعداد بیشتری از اتم‌های a، b، $\cdots$ هست، متناظر کردن هر عدد طبیعی n با دنباله‌ای است که شامل n رخداد از a است. بنابراین اعداد 1، 2، 3 و $\cdots$ به ترتیب با دنباله‌های a، $a * a$، $a * a * a$ و $\cdots$ تعریف می‌شوند. جمله «x یک عدد صحیح مثبت است» را با $T(x)$ نمایش می‌دهیم.
اگر x و y دو عدد صحیح مثبت باشند، «$x \sqsubseteq y$» مفهوم «$x \leqq y$» را نشان داده و «$x * y$» جمع x و y را نشان می‌دهد. ضرب «$x \times y$» دنباله‌ای است که از پشت سر هم قرار دادن x با خودش به تعداد y بار بدست می‌آید.

$$(\exists w)\Big\{w(y,z)\ \wedge\ (\forall s)(\forall t)\Big[w(s,t) \longrightarrow\ (s = a \wedge\ t = x) \vee (\exists u)(\exists v)(w(u,v) \wedge\ s = a * u \wedge\ t = x * v)\Big]\Big\} \qquad (۳.۴)$$

فرمول شماره (۳.۴) بیان می‌کند که دنباله w چنان وجود دارد که

$$w(y,z) \qquad (۴.۴)$$

و

$$(\forall s)(\forall t)\Big\{w(s,t) \longrightarrow (s=a\wedge\ t=x)\vee \quad (\text{۵.۴})$$
$$(\exists u)(\exists v)\big(w(u,v)\wedge\ s=a*u\wedge t=x*v\big)\Big\}$$

با (۴.۴) و (۵.۴) داریم:

$$.(y=a\wedge\ z=x)\ \vee(\exists u)(\exists v)[w(u,v)\wedge\ y=a*u\wedge\ z=x*v] \quad (\text{۶.۴})$$

حال اگر دنباله y شامل یک اتم باشد، از این‌رو هیچ الحاقی نداریم. جایگذاری در قسمت دوم فرمول (۶.۴) شکسته شده و در قسمت اول همان فرمول a به جای y و x به جای z آورده می‌شود. حال اگر y بیش از یک اتم داشته باشد، جایگذاری دوم در (۶.۴) بیان می‌کند که y و z به ترتیب با a و x نشان داده می‌شود و y' و z' باقی مانده‌ی y و z هست به‌طوری‌که:

$$.w(y',z') \quad (\text{۷.۴})$$

با (۵.۴) و (۷.۴) داریم:

$$.\Big(y'=a\ \wedge z'=x\Big)\ \vee(\exists u)(\exists v)\Big[w(u,v)\ \wedge y'=a*u\ \wedge z'=x*v\Big] \quad (\text{۸.۴})$$

در ادامه این ترتیب برای گام‌های بعدی مکان‌هایی برای دنباله اصلی y وجود دارد و خواهیم دید که در نهایت y فقط شامل a بوده و z شامل a که با تعدادی x الحاق شده است، می‌باشد. به‌طور خلاصه می‌توان گفت: y یک عدد صحیح مثبت است و

$$.z=x\times y \quad (\text{۹.۴})$$

حال عکس مطلب بالا را نشان خواهیم داد: برای هر عدد صحیح x و y، (۹.۴) بر (۳.۴) دلالت دارد. زوج مرتب‌های زیر را در نظر بگیرید.

$$\begin{array}{cc}
a & x\\
a*a & x*x\\
a*a*a & x*x*x\\
\vdots & \vdots\\
y & x\times y
\end{array}$$

(از مفهوم در نظر گرفته شده «$x \times y$» برمی‌آید که دنباله به‌دست آمده در ستون سمت راست در مقابل y، در حقیقت همان $x \times y$ می‌باشد). پس دنباله w وجود دارد به‌طوری‌که برای هر دنباله s و t، «$w(s,t)$» برقرار است اگر و تنها اگر s و t زوج مرتبی در لیست بالا باشند. از بررسی این فهرست مشاهده می‌شود که (۵.۴) برقرار است و همچنین

$$.w(y, x \times y) \qquad (۱۰.۴)$$

از (۹.۴) و (۱۰.۴) ما (۴.۴) را به‌دست می‌آوریم و از (۴.۴) و (۵.۴)، (۳.۴) را به‌دست می‌آوریم. بنابراین عمل $x \times y$ به عنوان یک و تنها یک دنباله z که شرط (۳.۴) را برآورده می‌کند، تعریف می‌شود.

$$D4. \qquad x \times y \equiv (\min z)(\exists w)\Big\{w(y,z) \wedge \ (\forall s)(\forall t)\Big[w(s,t) \longrightarrow \ (s = a \wedge t = x)$$
$$\vee(\exists u)(\exists v)(w(u,v) \wedge s = a * u \ \wedge t = x * v)\Big]\Big\}$$

توان «x^y» نیز مشابه ضرب «$x \times y$» تعریف می‌شود؛ با این تفاوت که ضرب «$x \times y$» دنباله‌ای است که از الحاق x با خودش به تعداد y بار به دست می‌آید و توان «x^y» دنباله‌ای است که از ضرب x با خودش به تعداد y بار به دست می‌آید.

$$D5. \qquad x^y \equiv (\min z)(\exists w)\Big\{w(y,z) \wedge \ (\forall s)(\forall t)\Big[w(s,t) \longrightarrow \ (s = a \wedge t = x)$$
$$\vee(\exists u)(\exists v)(w(u,v) \wedge s = a * u \ \wedge t = x \times v)\Big]\Big\}$$

تعاریف $D4$ و $D5$ برحسب دنباله‌های x و y که اعداد صحیح مثبت هستند، نوشته شده‌اند؛ ولی ممکن است x عدد صحیح مثبت نباشد اما y عدد صحیح مثبت است و ضرب «$x \times y$» شامل دنباله‌ای است که از الحاق x با خودش به تعداد y بار به دست می‌آید.

۵.۴ ساختار دو حرفی

حال برمی‌گردیم راجع به یک روش جدید برای تناظر اعداد صحیح مثبت با دنباله‌ها صحبت می‌کنیم. این بار از یک روش دو حرفی، به‌عبارت دیگر روشی که از تمامی دنباله‌ها استفاده می‌کند، استفاده می‌کنیم. برای ساختار دو حرفی در هر صورت لازم است بدانیم تعداد اتم‌ها چندتا هستند. ابتدا

ساختار برای حالتی که دقیقا دو تا اتم a و b داشته باشیم، تنظیم می‌شود. سپس باید ببینیم چگونه می‌توانیم روش را برای هر تعداد متناهی بزرگتر از ۲ اتم توسعه دهیم. یکسان‌سازی دو حرفی اعداد صحیح مثبت با دنباله‌های متناهی از اتم‌های a و b می‌تواند با مرتب‌سازی الفبایی دنباله‌ها صورت گیرد. به‌عبارت دیگر ابتدا طبق طول مرتب می‌کنیم و در مواردی که طول آن‌ها مساوی است الفبایی مرتب می‌کنیم:

a	b	aa	ab	ba	bb	aaa	aab	aba	abb	baa	bab	bba	bbb	...
۱	۲	۳	۴	۵	۶	۷	۸	۹	۱۰	۱۱	۱۲	۱۳	۱۴	...

در این روش نه‌تنها اعداد صحیح مثبت را با دنباله‌های متناهی می‌توان بیان کرد، برعکس نیز درست است. دنباله‌های متناهی از اتم‌های a و b را می‌توان به اعداد صحیح ترجمه کرد یعنی یک تناظر یک به یکی بین دنباله‌های متناهی و اعداد صحیح وجود دارد و عمل الحاق به عنوان یک عملگر حسابی روی اعداد صحیح مثبت به حساب می‌آید. در حقیقت عمل الحاق با اعمال حسابی مقدماتی به صورت زیر قابل بیان است:

$$x * y = y + [x \times (\min z)(\exists w)(z = 2^w \wedge z \leqq y + 1 < z + z)]$$

این کار ما دقیقا برعکس یافتن روش‌هایی برای تعریف $x+y$، $x \times y$ و x^y با استفاده از نظریه الحاق است. کار ما نشان دادن این است که عملگر جدید الحاق، یک عملگر حسابی است که شبیه به عملگرهای جمع و ضرب و توان می‌تواند تعریف شود. اگرچه دنباله‌های ساخته شده از رخدادهای a همان‌طور که در بخش ۴.۴ انجام داده شد شامل تمامی اعداد صحیح مثبت نبودند. آن‌ها نقش مهمی در اینجا بازی می‌کنند. تعاریف $D1$ تا $D3$ به قوت خود باقی خواهند ماند. تعاریف $D4$ و $D5$ که به عنوان تعاریف ضرب و توان اعداد صحیح مثبت در دسترس بودند همچنان برای حالات کمکی مفید خواهند بود. تعاریف $D4$ و $D5$ را با نمادهای بهبود یافته در زیر می‌بینیم.

$$D4'. \qquad x \times_\tau y \equiv (\min z)(\exists w)\Big\{w(y,z) \wedge (\forall s)(\forall t)\Big[w(s,t) \longrightarrow (s = a \wedge t = x) \vee (\exists u)(\exists v)(w(u,v) \wedge s = a * u \wedge t = x * v)\Big]\Big\}$$

$$D5'. \qquad x \uparrow_\tau y \equiv (\min z)(\exists w)\Big\{w(y,z) \wedge (\forall s)(\forall t)\Big[w(s,t) \longrightarrow (s = a \wedge t = x)$$
$$\vee(\exists u)(\exists v)(w(u,v) \wedge s = a * u \wedge t = x \times_\tau v)\Big]\Big\}$$

یک شمارش شامل n رخداد از a را از این به بعد یک n-برج خواهیم نامید. هر وقت x یک عدد صحیح مثبت باشد آن وقت x-برج را با τx نشان می‌دهیم. با توجه به راهنمایی‌های پایینی که در ادامه می‌آید x-برج در نظریه الحاق قابل تعریف است. در زیر لیستی از اعداد صحیح مثبت و متوالی از ۱ تا x را به همراه برج‌هایشان را می‌بینیم.

a	a
b	$a*a$
$a*a$	$a*a*a$
$a*b$	$a*a*a*a$
$b*a$	$a*a*a*a*a$
$b*b$	$a*a*a*a*a*a$
$a*a*a$	$a*a*a*a*a*a*a$
$a*a*b$	$a*a*a*a*a*a*a*a$
$\vdots$	$\vdots$
x	τx

روش تولید یک ستون در سمت چپ که شامل اعداد صحیح نرمال به ترتیب $1, 2, 3, \cdots$ باشد طبق قاعده‌های زیر توضیح داده شده است.

(i) با a و b شروع کن. (ii) اگر u در ستون ظاهر شد، یک $u*a$ را در یک مکان که بیش از دو برابر پایین‌تر از آن ستون باشد قرار بده و $u*b$ را بعد از آن قرار بده. به عبارت دیگر قرار بده: $u*a = 2u+1$ و $u*b = 2u+2$.

از (ii) همراه با طبیعت ستون سمت راستی (برج‌ها) مشهود است که هر جا u و v مقابل هم ظاهر

شدند به‌طوری‌که $v = \tau u$، آن چیزی که در مقابل $u * a$ ظاهر می‌شود $v * v * a$ است و آن چیزی که در مقابل $u * b$ ظاهر می‌شود $v * v * a * a$ است. با توجه به مشاهدات می‌توان تعاریف زیر را انجام داد:

$$D6. \qquad \tau x \equiv (\min y)(\exists w)\Big\{w(x,y) \wedge (\forall s)(\forall t)\Big[w(s,t) \longrightarrow (s = a \wedge t = a) \vee (s = b \wedge t = a * a) \vee ((\exists u)(\exists v)\big[w(u,v) \wedge (s = u * a \wedge t = v * v * a) \vee (s = u * b \wedge t = v * v * a * a)\big]\Big]\Big\}$$

حال تعاریف $x + y$، $x \times y$، و x^y بلافاصله به‌دست می‌آیند. ما $x + y$ را به صورت یک عدد صحیحی تعریف می‌کنیم که برج آن x-برج و y-برج را الحاق می‌کند و می‌توانیم $x \times y$ را به صورت یک عدد صحیح تعریف کنیم که برج آن یک برج ضرب (همان‌طور که در $D4'$ تعریف شد) از برج‌های x به تعداد y است؛ همینطور x^y را می‌توانیم تعریف کنیم.

$$D7. \qquad x + y \equiv (\min z)(\tau z = \tau x * \tau y)$$

$$D8. \qquad x \times y \equiv (\min z)(\tau z = \tau x \times \tau y)$$

$$D9. \qquad x^y \equiv (\min z)(\tau z = \tau x \uparrow \tau y)$$

روش‌های ساختاری که برای دو تا اتم a و b بیان شد، برای حالت کلی‌تر k تا اتم $a_1, \cdots, a_k$ نیز می‌تواند بسط داده شود.

۶.۴ اعداد طبیعی

در بخش‌های قبل دیدیم که چطور حساب مقدماتی اعداد صحیح مثبت می‌تواند توسط هر نظریه مقدماتی الحاق حداقل با دو اتم به‌دست آید. برای اعداد طبیعی از آنجا که 0 را داریم تناظری بدین ترتیب بین آنها و اعداد صحیح مثبت برقرار می‌کنیم:

0	1	2	3	4	5	...
1	2	3	4	5	6	...

اگر این تناظر را با $\mathcal{N}$ نشان دهیم می‌نویسیم:

$$0_{\mathcal{N}} = 1, \quad 1_{\mathcal{N}} = 2, \quad 2_{\mathcal{N}} = 3, \ \ldots$$

پس $x_{\mathcal{N}} = x + 1$. حال برای جمع $+_{\mathcal{N}}$ در اعداد طبیعی داریم:

$$(x_{\mathcal{N}} +_{\mathcal{N}} y_{\mathcal{N}}) = (x+1) +_{\mathcal{N}} (y+1) = x + y + 1 = (x+y)_{\mathcal{N}} \tag{۱۱.۴}$$

در این تعریف چون $x + 1$ با عدد x و عدد $y + 1$ با عدد y تناظر یک‌به‌یک دارند، بنابراین جمع این دو عدد $x + y$ می‌شود که در حساب اعداد طبیعی با $x + y + 1$ در تناظر است، به همین خاطر حاصل جمع $x + y + 1$ شده است. به‌طور مشابه می‌توانیم عمل «$x \times_{\mathcal{N}} y$» و همچنین عمل «$x \uparrow_{\mathcal{N}} y$» را روی علم حساب اعداد طبیعی تعریف کنیم:

$$\begin{aligned}(x_{\mathcal{N}} \times_{\mathcal{N}} y_{\mathcal{N}}) = (x+1) \times_{\mathcal{N}} (y+1) &= (x \times y) + (x + y) + 1 \\ &= (x \times y)_{\mathcal{N}}\end{aligned} \tag{۱۲.۴}$$

$$(x_{\mathcal{N}} \uparrow_{\mathcal{N}} y_{\mathcal{N}}) = (x+1) \uparrow_{\mathcal{N}} (y+1) = (x \uparrow y) + 1 = (x \uparrow y)_{\mathcal{N}} \tag{۱۳.۴}$$

هر جا که «$(x \times y)$» «$(x \uparrow y)$» را داشته باشیم منظورمان به‌ترتیب همان عمل ضرب و عمل توان را روی حساب اعداد صحیح مثبت است. فرمول‌های (۱۱.۴)، (۱۲.۴) و (۱۳.۴) برای تعاریف عمومی $+_{\mathcal{N}}$، $\times_{\mathcal{N}}$، و $\uparrow_{\mathcal{N}}$ کافی نیستند. زیرا تعاریف، این اعمال را در فرم $(x + 1)$ و $(y + 1)$ برای اعداد توصیف می‌کنند و یک عدد وجود دارد که 1 نامیده شده (همان $0_{\mathcal{N}}$) و نمی‌تواند به فرمی که گفته شد نمایش داده شود. بنابراین باید فرمول‌های (۱۱.۴)، (۱۲.۴) و (۱۳.۴) را تکمیل کنیم.

اعمالی که روی $0_\mathcal{N}$ تعریف می‌شود به صورت زیرند:

$$0_\mathcal{N} +_\mathcal{N} z = z \qquad (۱۴.۴)$$

$$z +_\mathcal{N} 0_\mathcal{N} = z \qquad (۱۵.۴)$$

$$0_\mathcal{N} \times_\mathcal{N} z = 0_\mathcal{N} = 1 \qquad (۱۶.۴)$$

$$z \times_\mathcal{N} 0_\mathcal{N} = 0_\mathcal{N} = 1 \qquad (۱۷.۴)$$

$$0_\mathcal{N} \uparrow_\mathcal{N} (y+1) = 0_\mathcal{N} = 1 \qquad (۱۸.۴)$$

$$z \uparrow_\mathcal{N} 0_\mathcal{N} = 1_\mathcal{N} = 2 \qquad (۱۹.۴)$$

بالاخره تعاریف کلی زیر را خواهیم داشت:

$$.x +_\mathcal{N} y \equiv (\min z)(x + y = z + 1)$$

$$.x \times_\mathcal{N} y \equiv (\min z)[(x+y) + z = (x \times y) + 2]$$

$$.x \uparrow_\mathcal{N} y \equiv (\min z)[(y = 1 \wedge z = 2) \vee (y \neq 1 \wedge x = z = 1) \vee$$

$$\exists w \ (x = w + 1 \wedge z \times w = w^y + w)])$$

مراجع

[1] Egon Börger, Erich Gradel, and Yuri Gurevich. *The Classical Decision Problem.* Springer, Berlin, 1997.

[2] Ian Chiswell, *A Course in Formal Languages, Automata and Groups.* Queen Mary, University of London, 2000.

[3] Herbert B. Enderton, A Mathematical Introduction to Logic. University of California, Los Angeles, 2nd edition, 2001.

[4] Yuri Gurevich, The Decision Problem for the Logic of Predicates and Operations. *Algebra i Logika* (1969) 284–308.English translation: *Algebra and Logic* 8 (1969) 160–174.

[5] Daniel Leivant, Logical Undecidabilities Made Easy. *Fundamenta Informaticae* 89 (2008) 307–312.

[6] Willard V. Quine, Concatenation as a Basis for Arithmetic. *Journal of Symbolic Logic* 11 (1946) 105–114.

واژه‌نامه فارسی به انگلیسی

اتم .... Atom
استقراء .... Induction
استنتاج .... Deduction
اشتقاق .... Derivation
الحاق .... Concatenation
الفبا .... Alphabet
بازگشتی .... Recursive
برهان .... Proof
پیکربندی .... Configuration
تابع الحاق .... Concatenation Function
تابع بازگشتی .... Recursive Function
تابع دوتایی .... Binary Function
تابع همانی .... Identity Function
تالی .... Successor
انتقال .... Transition
تراگذاری .... Transitive
تصمیم‌ناپذیری .... Undecidablity
تعریف‌پذیری .... Definability
تك موضعی .... Unary
تناظر یك به یك و پوشا .... Bijection
تورینگ- کامل .... Turing-Complate
حساب .... Arithmetic

حل‌ناپذیر	Unsolvable
دستور زبان	Grammar
دنباله	Sequence
رابطه دوتایی	Binary Relation
رابطه یک تایی	Unary Relation
رخداد	Occurrence
رشته	String
زبان	Language
زبان صوری	Formal Language
زیر رشته	Sub-String
ساختار دوجزئی	Bilateral Construction
ساختار یک جزئی	Unilateral Construction
سور	Quantifier
شرکت‌پذیری	Associativity
شمارش‌پذیر	Countable
شمارش‌پذیر کارآمد	Effectively Countable
عملگر	Operator
قضیه	Theorem
کدگذاری	Encoding
ماشین تورینگ	Turing Machine
محاسبه‌پذیر	Computable
مسئله توقف	Halting Problem
مقدماتی	Elementary
منطق مرتبه اول	First-Order Logic
میان‌وند	Infix
ناتمامیت	Incompleteness
نتیجه	Conclusion
نماد رابطه‌ای	Relation Symbal
نیم تصمیم‌پذیر	Semi-Decidable
نیم دستور زبان	Semi-Grammar
همانی	Identity

واژه‌نامه انگلیسی به فارسی

Alphabet	الفبا
Arithmetic	حساب
Associativity	شرکت‌پذیری
Atom	اتم
Bijection	تناظر یك به یك و پوشا
Bilateral Construction	ساختار دوجزئی
Binary Function	تابع دوتایی
Binary Relation	رابطه دوتایی
Computable	محاسبه‌پذیر
Concatenation Function	تابع الحاق
Countable	شمارش‌پذیر
Concatenation	الحاق
Conclusion	نتیجه
Configuration	پیکربندی
Deduction	استنتاج
Definability	تعریف‌پذیری
Derivation	اشتقاق
Effectively Countable	شمارش‌پذیر کارآمد
Elementary	مقدماتی
Encoding	کدگذاری
First-Order Logic	منطق مرتبه اول
Formal Language	زبان صوری

Grammar .... دستور زبان
Halting Problem .... مسئله توقف
Identity .... همانی
Identity Function .... تابع همانی
Incompleteness .... ناتمامیت
Induction .... استقراء
Infix .... میان‌وند
Language .... زبان
Occurrence .... رخداد
Operator .... عملگر
Proof .... برهان
Quantifier .... سور
Recursive .... بازگشتی
Recursive Function .... تابع بازگشتی
Relation Symbal .... نماد رابطه‌ای
Semi-Decidable .... نیم تصمیم‌پذیر
Semi-Grammar .... نیم دستورزبان
Sequence .... دنباله
String .... رشته
Sub-String .... زیررشته
Successor .... تالی
Theorem .... قضیه
Transition .... انتقال
Transitive .... تراگذاری
Turing-Complate .... تورینگ- کامل
Turing Machine .... ماشین تورینگ
Unary .... تك موضعی
Unary Relation .... رابطه یك تایی
Undecidablity .... تصمیم‌ناپذیری
Unilateral Construction .... ساختار یك جزئی
Unsolvable .... حل‌ناپذیر

نمایه

Surname: Sadeghi bigham **Name:** Shokoufeh

Title: Undecidability Of First-Order Logic

Supervisor: Saeed Salehi

Advisor: Mohammad Shahryari

Degree: Master of Science **Subject:** Pure Mathematics

Field: Mathematical Logic **University of Tabriz**

Faculty of Mathematical Sciences **Date:** 2014 **Number of Pages:** 41

Keywords: Undecidability, Semi-undecidability, Incompleteness.

Abstract

Referring to symbolic computing permits extremely simple proofs of undecidability for first-order logic and theories of basic structures. Using grammars we obtain a trivial proof that first-order validity is undecidable (already for formulas with quantifier prefix ∃∃ and referring to only one unary relation and one binary function). A similar proof yields Gödel's First Incompleteness Theorem for the structure of strings; undecidability for arithmetic follows by noting that addition and multiplication permit direct coding of string concatenation.

University of Tabriz

Faculty of Mathematical Sciences

DISSERTATION SUBMITTED IN PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR THE DEGREE OF MASTER OF SCIENCE IN PURE MATHEMATICS (MATHEMATICAL LOGIC)

Undecidability Of First-Order Logic

Supervisor

Saeed Salehi

Advisor

Mohammad Shahryari

by

Shokoufeh Sadeghi bigham

2014