درس مبانی نظریه محاسبه

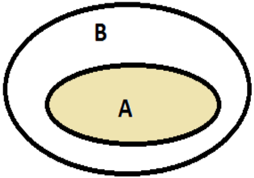
جلسه بیست و دوم

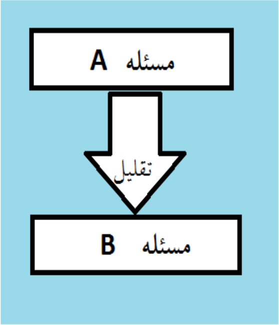
چند مسئله تصمیم ناپذیر

A few undecidable problems

مروری بر روش تقلیل Reduction Method

تعریف: مسئله A را تقلیل‌ـپذیر به مسئله B گویند اگر با یک الگوریتم برای B بتوان مسئله A را حل کرد. اینطور نشان داده‌ایم که مسئله B حداقل به سختی مسئله A است.





اگر با یک الگوریتم برای B

بتوان مسئله A را حل کرد

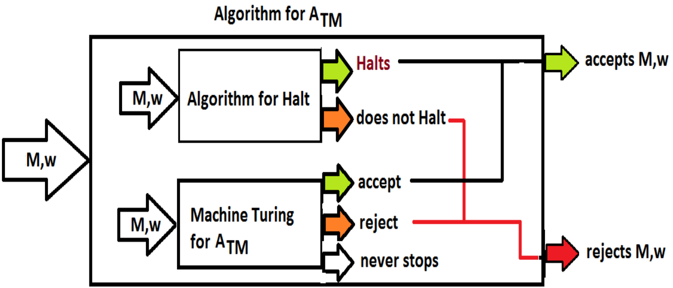
مسئله توقف ماشین تورینگ

$Halt =$

$\{\langle M, w\rangle \mid$ یک ماشین تورینگ است که روی رشته w متوقف می‌شود $M\}$

قضیه: مسئله $Halt$ تصمیم ناپذیر است.

اثبات: از روش تقلیل استفاده می‌کنیم. نشان می‌دهیم اگر الگوریتمی برای $Halt$ وجود داشته باشد، آنگاه مسئله A_{TM} را می‌توان حل کرد. (مسئله A_{TM} را به مسئله $Halt$ تقلیل می‌دهیم).

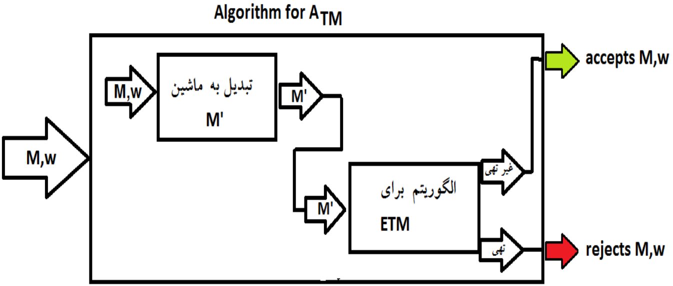


مسئله تهی بودن زبان ماشین تورینگ

$$E_{\mathsf{TM}} = \{\langle M\rangle \mid M \text{ is a TM and } L(M) = \emptyset\}$$

قضیه: E_{TM} تصمیم ناپذیر است.

اثبات: از روش تقلیل استفاده می‌کنیم. نشان می‌دهیم اگر الگوریتمی برای E_{TM} وجود داشته باشد، آنگاه مسئله A_{TM} را می‌توان حل کرد. به عبارت دیگر، مسئله A_{TM} را به مسئله E_{TM} تقلیل می‌دهیم.



با داشتن زوج ورودی M و w می‌خواهیم مشخص کنیم که آیا ماشین M رشته w را می‌پذیرد یا نه (مسئله A_{TM})

برای این کار، ابتدا از روی توصیف M و w ماشین تورینگ M' را می‌سازیم بطوریکه

$$\begin{cases} L(M') \neq \emptyset & \text{if } M \text{ accepts } w \\ L(M') = \emptyset & \text{if } M \text{ does not accept } w \end{cases}$$

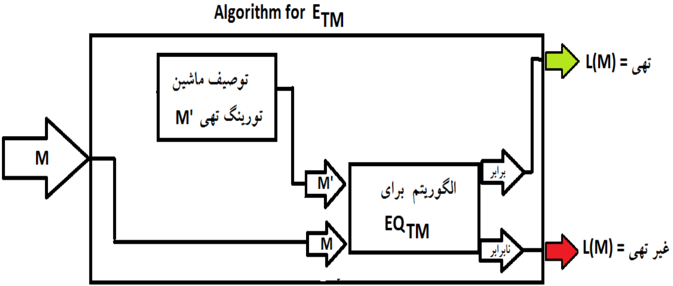
توصیف ماشین M': ماشین تورینگ M' اول چک می‌کند، اگر رشته ورودی چیزی غیر از w باشد به q_{reject} می‌رود. اگر رشته ورودی w باشد، ماشین M را روی رشته w اجرا می‌کند و نتیجه هر چه بود گزارش می‌کند.

مسئله تساوی زبان دو ماشین تورینگ

$$EQ_{\mathsf{TM}} = \{\langle M_1, M_2\rangle |\ M_1 \text{ and } M_2 \text{ are TMs and } L(M_1) = L(M_2)\}$$

قضیه: EQ_{TM} تصمیم ناپذیر است.

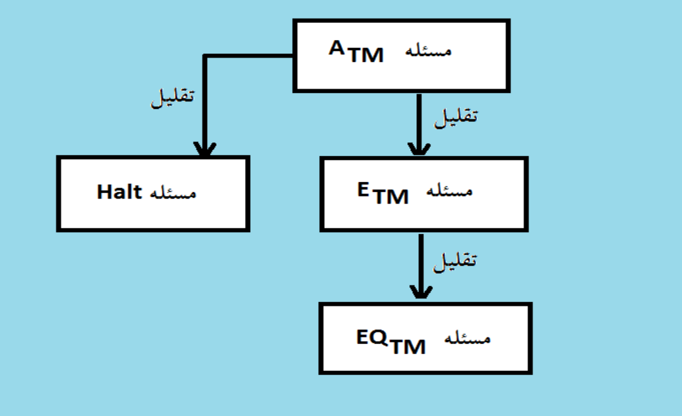
اثبات: از روش تقلیل استفاده می‌کنیم. نشان می‌دهیم اگر الگوریتمی برای EQ_{TM} وجود داشته باشد، آنگاه مسئله E_{TM} را می‌توان حل کرد. به عبارت دیگر، مسئله E_{TM} را به مسئله EQ_{TM} تقلیل می‌دهیم.



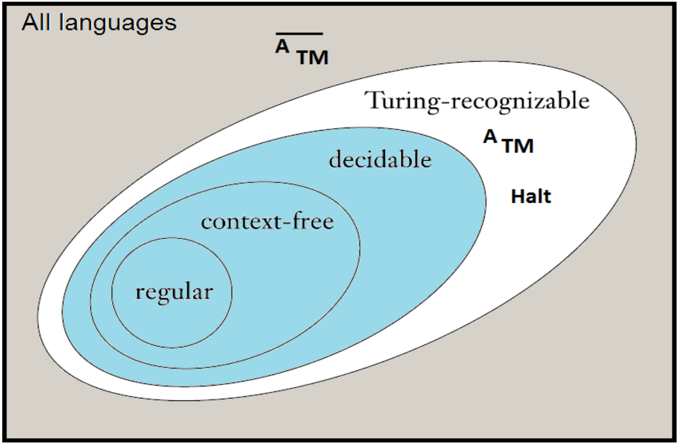
با داشتن ورودی M می‌خواهیم مشخص کنیم که آیا ماشین M زبانش تهی است یا نه (مسئله E_{TM})

برای این کار، ابتدا یک ماشین تورینگ که زبانش تهی باشد را می‌سازیم. نام این ماشین را M' می‌گذاریم. ساختن چنین ماشینی خیلی راحت است. این ماشین همه چیز را رد می‌کند.

حال ماشین M' و ماشین M را به الگوریتم EQ_{TM} می‌دهیم. اگر زبان M تهی باشد، زبان M و M' برابر خواهد بود در غیر این صورت زبان این دو ماشین برابر نخواهد بود.


مسئله A_{TM}
تقليل
تقليل
مسئله Halt
مسئله E_{TM}
تقليل
مسئله EQ_{TM}

سلسله مراتب زبانها



زبانهای

E_{TM}

EQ_{TM}

کجای سلسله مراتب هستند؟

تمرین: نشان دادیم زبان E_{TM} تصمیم ناپذیر است. نشان دهید این زبان Turing-recognizable هم نیست.

یک مسئله تصمیم ناپذیر ساده

تعریف: یک دومینو بصورت یک کسر است که در صورت و مخرج کسر آن یک رشته قرار گرفته است. مانند نمونه زیر

$$[\frac{a}{ab}]$$

مسئله تناظر پُست: فرض کنید D مجموعه‌ای متناهی از دومینوها باشد. می‌خواهیم بدانیم آیا با قرار دادن تعدادی از دومینوهای داخل D (با فرض اینکه بتوانیم آنها را تکرار هم بکنیم) آیا می‌توانیم یک سلسله از دومینوها ایجاد کنیم که رشته بالا و پایین کسر یکسان باشند؟ به این سلسله از دومینوها یک تناظر گفته می‌شود.

مثال:

$$D = \{[\frac{b}{ca}], [\frac{a}{ab}], [\frac{ca}{a}], [\frac{abc}{c}]\}$$

جواب مسئله تناظر پُست در مورد مجموعه دومینوهای بالا مثبت است. سلسله دومینوهای زیر یک تناظر را ایجاد کرده است.

$$\left[\frac{a}{ab}\right]\left[\frac{b}{ca}\right]\left[\frac{ca}{a}\right]\left[\frac{a}{ab}\right]\left[\frac{abc}{c}\right]$$

دقت کنید برای ایجاد یک تناظر مجاز هستیم یک دومینو را تکرار بکنیم. در تناظر بالا دومینوی $\left[\frac{a}{ab}\right]$ تکرار شده است.

قضیه: مسئله تناظر پُست یک مسئله تصمیم ناپذیر است. به عبارت دیگر، با داشتن مجموعه‌ای از دومینوها هیچ الگوریتمی وجود ندارد که تشخیص دهد آیا می‌توان یک تناظر با این مجموعه از دومینوها ساخت یا نه.