

نقش و کاربرد گرامرهای ویژه در مهندسی معکوس نرم افزار

**تمرین (2) درس کامپایلر پیشرفته نیمسال 1-1395**

دانشجو:

**مرتضی ذاکری**

استاد:

**دکتر سعید پارسا**

**پاییز 1395**

# مقدمه

در این نوشتار به صورت خلاصه به کاربردها و نحوه استفاده از *گرامرهای ویژه*[[1]](#footnote-1) (گرامرهای دارای صفت (رجوع شود به مبحث کامپایلرها)، در مباحث مربوط به *مهندسی معکوس نرم افزار*[[2]](#footnote-2) می پردازیم. ابتدا مفاهیم گرامرهای ویژه و مهندسی معکوس را شرح می دهیم و سپس چکیده چند مقاله و گزارش کاربردی مرتبط در این زمینه را مرور می کنیم.

# مفاهیم پایه

## گرامر ویژه

طبق کتاب [1] در تعریف گرامر ویژه هر قانون تولید گرامر $A \rightarrow α$ متناظر با یک مجموعه از قوانین معنایی به صورت $b≔f(c\_{1}. c\_{2}. …. c\_{k})$ که در آن $f$ یک تابع است و $b$:

1. $b$ یک صفت اکتسابی از A و $c\_{1}. c\_{2}. …. c\_{k}$ صفت هایی هستند که متعلق به نماد های گرامر قانون تولید هستند و یا
2. $b$ یک صفت موروثی از یکی از نمادهای گرامر در طرف راست قانون تولید است و $c\_{1}. c\_{2}. …. c\_{k}$ صفت هایی متعلق به نمادهای گرامر هستند.

پایانه ها بنا به فرض فقط دارای صفت های اکتسابی هستند، که معمولا توسط تحلیل گر لغوی فراهم می شود. غیر پایانه ها دارای صفت های موروثی نیز هستند که در درخت تجزیه عبارت ظاهر می شود.

دو طرح و استفاده غالب برای گرامر ویژه در کامپایلر مطرح می گردد. یکی در ایجاد صفات مربوط به آزمون نوع در فاز تحلیل معنایی برنامه و دیگری ایجاد صفات لازم برای تولید کد در مرحله تحلیل نحوی (ترجمه هدایت شده یا مبتنی بر دستور). البته این فاز ها معمولا همزمان انجام می پذیرد. به عبارت ساده در گرامرهای ویژه برای هر غیرپایانه یا متغیر میانی یک صفت از جنس رشته در نظر گرفته می شود که محتوای آن حاوی **کدی** است که برای متغیر میانی ایجاد می شود.

## مهندسی معکوس

واژه مهندسی معکوس نخستین بار از در فناوری های سخت افزاری ظاهر شد و به فرایند اقتباس ویژگی ها و ساختار یه سخت افزار پیچیده اشاره می کند. اخیرا این اصطلاح در حوزه نرم افزار نیز وارد شده است که اگر چه معنایی مشابه دارد اما هنوز تعریف دقیقی از آن ارائه نشده است. طبق [2] مهندسی معکوس عبارت است از:

 «فرایند تحلیل یک سیستم به منظور شناسایی اجزای سیستم و روابط داخلی حاکم بر آن ها، و ایجاد یک نمایش از سیستم در شکل دیگر و سطح بالاتری از انتزاع.»

#  کاربردها

## آیا گرامرهای ویژه در صنعت کاربرد دارد؟

گرامرهای ویژه هم در مطالعه های آکادمیک و هم در کارهای صنعتی کاربرد دارند. اما به جنبه دوم توجه کمتری شده است زیرا اغلب افراد درحوزه کامپیوتر از آن بی اطلاع هستند. اجازه دهید ببینیم این ساختار ها کجا و چگونه کاربرد دارند؟

یکی از کمپانی هایی که در زمینه استفاده از این گرامرها فعالیت دارد کمپانی آلمانی **CoCoLab** است[[3]](#footnote-3). مهمترین زمینه تجارتی این شرکت پارسرها یا front-end ها برای زبان هایی مثل COBOL، PL/1، C، C++ و جاوا است. این پارسرها معمولا برنامه های تحلیل گر در پروژه های مهندسی معکوس کاربرد دارند. در این پارسرها گرامرهای ویژه برای دو وظیفه به کار می روند. نخست، ساخت درخت تجزیه و دیگر درخت ها در حین عملیات تجزیه. دوم، برای تحلیل نام در تحلیل مفهومی که بر روی درخت نحوی مجازی انجام می شود. توسعه گرامرهای ویژه برای زبان های مختلف زمان بر است و بین یک روز تا یک یا دو ماه طول می کشد. این زمان بستگی به اندازه و پیچیدگی گرامر دارد. برای مثال گرامر زبان COBOL تقریبا 3000 قانون دارد. با این حال این عمل یک بار انجام می شود و سپس قابل استفاده روی تمام کدهای نوشته شده به آن زبان خواهد بود [3].



شکل 1 نمونه ای از گرامر ویژه ساخت درخت تجزیه نحوی در C++

گرامر ویژه فوق یک نمونه کاربرد کوچک در روی گرامر زبان C++ را نشان می دهد. دو صفت در قواعد این گرامر محاسبه می شوند. صفت tree که برای ساخت درخت تحلیل نحوی به کار می رود. صفت type که نوع هر عبارت را محاسبه می کند.

## ابزارهای خاص کمپانی CoCoLab

برخی دیگر از کارکردهایی که این کمپانی با استفاده از گرامرهای ویژه ایجاد کرده است عبارتند از:

* **انتقال درخت در زبان PL/1[[4]](#footnote-4):** در این کاربرد که مخصوص مهندسی معکوس کد در زبان PL/1 هست، تشخصی فراخوانی تابع به صورت F(X) حل شده است. در این زبان F(X) بسته به تعریف F هم نشان دهنده فراخوانی تابع است و هم نشان دهنده دسترسی به عناصر آرایه. با توجه به این که PL/1 اجازه استفاده قبل از تعریف را می دهد، تحلیل گر نحوی ممکن است تعریف F را هنگام ساخت درخت تجزیه متوجه نشود. بنابراین، پارسر این متغیر را به یک گره درخت در اولین مکان ممکن نگاشت می کند. بعدا یک انتقال درخت تجزیه صورت می گیرد و نتایج تحلیل مفهومی مشخص می کند که نوع کاربرد متغیر نام برده نهایتا چیست. این انتقال به صورت گرامرهای ویژه و با استفاده از یک برنامه برنامه پردازش گر درخت به نام puma نوشته شده است.
* **اعتبار سنجی اسناد [[5]](#footnote-5)XML:** با استفاده از گرامر های DTD که نوع محدود تری از گرامرهای آزاد یا مستقل از متن است. می توان یک سند XML را اعتبارسنجی کرد. این عمل به طبقه بندی اسناد xml کمک می کند. مشابه مورد قبلی این کار به وسیله ترکیب استفاده از گرامر های ویژه و پردازش گر درخت puma قابل انجام است.
* **الگوریتم نمایه[[6]](#footnote-6):** در این کارکرد ابزار مدیریت درخت ast از بسته نرم افزاری Cocktail یک شمای گرافیکی از گراف ها و درخت ها که از کد برنامه استخراج شده اند را نشان می دهد. گرامر ویژه نمایش داده شده در زیر بخشی از گرامر لازم برای انجام این کار است.



شکل 2 گرامر ایجاد گرافیکی درخت تجزیه



شکل 3 درخت تجزیه حاصل از مهندسی معکوس کد

## مهندسی معکوس مبتنی بر گرامر برای تایید رفتار و تست نرم افزار

در مقاله [4] روشی با استفاده از گرامر های گراف برای بررسی و تایید رفتار سیستم های با بی درنگ یا تضمین شده[[7]](#footnote-7) شرح داده شده است. در این مقاله روشی شرح داده شده است که با استفاده از گرامر، متن برنامه را به صورتی که زبان های ویژوال و نمودار های معمول UML نشان می دهند، بازنمایی می کند. به این ترتیب نحوه بررسی رفتار چنین سیستمی در مدل جدید بسیار آسان تر خواهد بود.

# نتیجه گیری

با توجه به موارد گفته شده می توان گفت امروزه ضرورت استفاده از تکنیک های موجود مباحث مربوط به کامپایلر، در مهندسی معکوس نرم افزار کاملا محسوس است. با این حال وسعت کاربرد بسیاری از این تکنیک ها به دلیل عدم وجود ابزارهای قوی و شناخته شده، همچنان اندک است و بیش تر توجه های صورت گرفته در حوزه آکادمیک هستند تا حوزه تجاری. گرامرهای ویژه نیز در این میان نقش منحصر به فردی داشته و به کمک آن می توان تقریبا هر سیستم نرم افزاری را که کد قابل دسترسی دارد، مهندسی معکوس نمود. توسعه ابزارهای مهندسی معکوس مبنی بر گرامرهای ویژه مبحثی آکادمیک و در عین حال کاربردی و جذاب خواهد بود.

# منابع

|  |  |
| --- | --- |
| [1]  | V. Aho, R. Sethi and D. Ullman, Compilers: Principles, Techniques and Tools, US: Pearson, 2014.  |
| [2]  | M. v. d. Brand , P. Klint and C. Verhoef, "Re-engineering needs Generic Programming Language Technology," p. 17, 1996.  |
| [3]  | J. Grosch, "Are Attribute Grammars Used in Industry?," CoCoLab - Datenverarbeitung, Germany. |
| [4]  | C. Zhao and K. Zhang , "A Grammar-Based Reverse Engineering Framework for Behavior Verification".  |

1. Attributed Grammars [↑](#footnote-ref-1)
2. Software Reverse Engineering [↑](#footnote-ref-2)
3. [www.cocolab.de](http://www.cocolab.de) [↑](#footnote-ref-3)
4. Tree Transformation in PL/I [↑](#footnote-ref-4)
5. Validation of XML Documents [↑](#footnote-ref-5)
6. Layout Algorithm [↑](#footnote-ref-6)
7. High Assurance System [↑](#footnote-ref-7)