

بسمه تعالی

تکلیف سوم درس طراحی کامپایلر (ترجمه مبتنی بر نحو-تولید کدمیانی و بهینه‌سازی)
در مورد تکلیف می‌توانید با دوستان خود مشورت کنید ولی تکلیف باید کار خود شما باشد.

۱- تعریف مبتنی بر نحو زیر را در نظر بگیرید.

الف) این تعریف (Lattributed, S-attributed یا هیچ کدام) است؟ چرا؟

ب) این تعریف را به نحوی به طرح ترجمه مبتنی بر نحو تبدیل کنید که قابل پیاده‌سازی به وسیله پارسرهای پایین به بالا باشد.

Production	Semantic Rules
$S \rightarrow B$	$B.ps = 10$ $S.ht = B.ht$
$B \rightarrow B_1 B_2$	$B_1.ps = B.ps$ $B_2.ps = B.ps$ $B.ht = \max(B_1.ht, B_2.ht)$
$B \rightarrow B_1 \text{ sub } B_2$	$B_1.ps = B.ps$ $B_2.ps = 0.7 * B.ps$ $B.ht = \max(B_1.ht, B_2.ht - 0.25 * B.ps)$
$B \rightarrow \text{text}$	$B.ht = \text{getHt}(B.ps; \text{text.lexval})$

۲-

الف) گرامر مستقل از متنی بنویسید که به وسیله آن بتوان رشته‌های باینری روی الفبای $\{0,1\}$ ایجاد کرد، به قسمی که رشته‌های ایجاد شده نمایش‌دهنده اعداد مضرب ۴ باشند. (۰ را به عنوان مضرب ۴ در نظر بگیرید و داشتن اعداد بدین شکل "00100" که در آن ۰ در انتهای عدد است، ایرادی ندارد.)

ب) برای گرامر مستقل از متن ایجاد شده، قوانین معنایی را به نحوی بنویسید که در نهایت در یکی از attr. های تعریف شده برای نماد شروع گرامر، مقدار عددی که به صورت باینری نمایش داده شده است، قرار بگیرد. برای مثال برای رشته "00100"، نماد شروع دارای attr. ای باشد که در نهایت مقدار ۴ در آن قرار بگیرد.

ج) با استفاده از attribute grammar نوشته شده در مثال قبل، annotated parse tree را برای رشته "11100" رسم کرده و وابستگی بین ارزیابی attr. ها را در این درخت نشان دهید.

۳- گرامر زیر را برای یک زبان ساده در نظر بگیرید.

$\text{stmt_list} \rightarrow \text{stmt_list stmt} \mid \text{stmt}$

$\text{stmt} \rightarrow \text{id} = \text{expr}$

$\text{expr} \rightarrow \text{expr} + \text{expr} \mid \text{expr} \text{ or } \text{expr}$

$\text{expr} \rightarrow \text{id} \mid \text{num}$

$\text{expr} \rightarrow \text{true} \mid \text{false}$

که در آن id توکنی است که یک شناسه (identifier) را مشخص میکند. num توکنی است که یک ثابت عددی را مشخص میکند و true و false دو ثابت بولی هستند. فرض کنید، id.type attr نوع یک id را مشخص میکند که میتواند "int" یا "bool" باشد.

قواعد کنترل نوع برای این زبان بدین صورت تعریف شده است که:

- هر دو عملوند عملگر جمع باید دارای نوع "int" باشند.
- هر دو عملوند عملگر or باید از نوع "bool" باشند.
- هر دو طرف عملگر انتساب باید از یک نوع باشند.

یک تعریف مبتنی بر نحو برای کنترل نوع و تولید کدمیانی برای تمامی عبارت های قابل تولید توسط گرامر بالا ارائه دهید.

۴- تعریف مبتنی بر نحوی بنویسید که کدمیانی برای for ایجاد کند.

$S \rightarrow \text{for} (S_1; B; S_2) S_3$

۵- دستورات سه آدرسه بلوک پایه زیر را به شکل یک DAG نمایش دهید. بر اساس تبدیل حذف زیرعبارت مشترک کدام یک از دستورات حذف خواهند شد؟

$t_1 = j * k$
 $t_2 = i + t_1$
 $t_3 = j * k$
 $t_4 = t_2 + t_3$
 $x = t_4$
 $j_5 = j * k$
 $t_6 = j + k$
 $t_7 = t_5 / t_6$
 $y = t_7$
 $t_8 = -y$
 $t_9 = t_8 * y$
 $t_{10} = j * k$
 $m = t_{10}$