

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

(الف)

attribute: B.ht

attribute: B.ps

این نوع L-attributed است. زیرا مقدار attribute {S موردش از Attr. غیر پایانی هم سمت چپ
Attr. موردش غیر پایانی سمت راست است آمده است.

(ب)

$S \rightarrow M_1 B \quad \{S.ht = B.ht\}$

$M_1 \rightarrow \epsilon \quad \{M_1.s = M_1.e\} \quad M_1.e = 10$

$B \rightarrow M_2 B_1 M_3 B_2 \quad \{B.ht = \max(B_1.ht, B_2.ht)\}$

$M_2 \rightarrow \epsilon \quad \{M_2.s = M_2.e\} \quad M_2.e = B.ps$

$M_3 \rightarrow \epsilon \quad \{M_3.s = M_3.e\} \quad M_3.e = B.ps$

$B \rightarrow M_4 B_1 \text{ sub } M_5 B_2 \quad \{B.ht = \max(B_1.ht, B_2.ht - 0.25 * B.ps)\}$

$M_4 \rightarrow \epsilon \quad \{M_4.s = M_4.e\} \quad M_4.e = B.ps$

$M_5 \rightarrow \epsilon \quad \{M_5.s = M_5.e\} \quad M_5.e = 0.7 * B.ps$

$B \rightarrow \text{text} \quad \{B.ht = \text{getHt}(B.ps, \text{text.lexval})\}$

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 0 در انتهای عدد است، ایرادی ندارد.)

$$S \rightarrow A100C$$

$$C \rightarrow C0 \mid \varepsilon$$

$$A \rightarrow A0 \mid A1 \mid \varepsilon$$

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.

$$S \rightarrow A100C \quad \{ S.val = A.val * 2^{(C.t+3)} + 2^{(C.t+2)} \}$$

$$C \rightarrow C_10 \quad \{ C.t = C_1.t + 1 \}$$

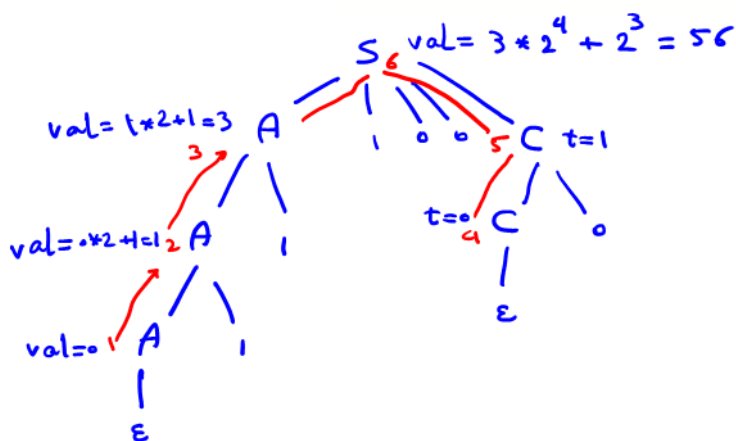
$$C \rightarrow \varepsilon \quad \{ C.t = 0 \}$$

$$A \rightarrow A_10 \quad \{ A.val = A_1.val * 2 \}$$

$$A \rightarrow A_11 \quad \{ A.val = A_1.val * 2 + 1 \}$$

$$A \rightarrow \varepsilon \quad \{ A.val = 0 \}$$

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.



stmt_list \rightarrow stmt_list stmt | stmt

stmt \rightarrow id = expr

expr \rightarrow expr + expr | expr or expr

expr \rightarrow id | num

expr \rightarrow true | 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.

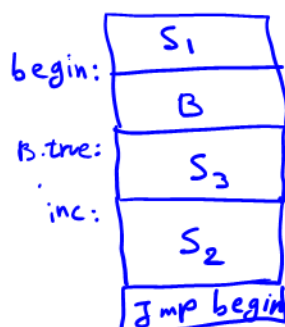
یک تعریف مبتنی بر نحو برای کنترل نوع و تولید کد میانی برای تمامی عبارت های قابل تولید توسط گرامر بالا ارائه دهید.

Semantic Rules

stmt_list \rightarrow stmt_list, stmt	stmt_list.code = stmt_list.code stmt.code
stmt_list \rightarrow stmt	stmt_list.code = stmt.code
stmt \rightarrow id = expr	if (id.type != expr.type) error else stmt.code = expr.code gen(top.get(id.lex) = 'expr.addr')
expr \rightarrow expr ₁ + expr ₂	expr.addr = Temp() if (expr ₁ .type != 'int' expr ₂ .type != 'int') error else expr.code = expr ₁ .code expr ₂ .code gen(expr.addr = 'expr ₁ .addr + 'expr ₂ .addr)
expr \rightarrow expr ₁ or expr ₂	expr.addr = Temp() if (expr ₁ .type != 'bool' expr ₂ .type != 'bool') error else expr.code = expr ₁ .code expr ₂ .code gen(expr.addr = 'expr ₁ .addr 'or' expr ₂ .addr)
expr \rightarrow id	expr.addr = top.get(id.lexem) expr.type = id.type expr.code = ''
expr \rightarrow num	expr.addr = num.val expr.type = 'int'
expr \rightarrow true	expr.addr = '1' expr.type = 'bool'

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

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



B.false:

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

```
begin = new Label()
inc = new Label()
B.true = new Label()
B.false = S.next
S3.next = inc
S2.next = begin
S.code = S1.code || Label(begin) B.code ||
Label(B.true) S3.code || Label(inc) S2.code ||
gen('jmp' begin)
```

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

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

