

言語処理工学 A 中間テスト

2014年12月19日

井上克郎

ノート教科書持ち込みなし。[1]は解答用紙表紙、[2]は折りたたんだ内側2枚に、[3]は裏表紙に解答を書くこと。間違った場所を書いた場合は減点する。

[1] 次の文法 G について答えよ。(30点) (1ページに答えをかけ)

- ① 左くりだしし、その左再帰性を除去した文法 G' を作れ。
- ② G' の各非終端記号の **FIRST** 集合を求めよ。
- ③ G の各非終端記号に対して、等価な文を生成する構文図を書け。

G :
 $E \rightarrow E+T \mid E-T \mid T$
 $T \rightarrow T \div F \mid T \times F \mid F$
 $F \rightarrow i$

[2] 次の拡張文法 G にたいして答えよ。(2,3ページにかけ) (10点×5)

- ① 各非終端記号の **Follow** 集合を求めよ。
- ② 正準 **LR(0)** 集合を求めて、**LR(0)** オートマトンの表を書け。
- ③ **LR** 構文解析表を求めよ。(縦に状態 $0, 1, \dots$, 横に記号 $i, /, *, \$, E', E$ をこの順に書くこと)。
- ④ $i / i * i$ の構文解析の過程を示せ。
- ⑤ それによって得られる構文木を示せ。

文法 G : (0) $E' \rightarrow E$

(1) $E \rightarrow i / E$

(2) $E \rightarrow i * E$

(3) $E \rightarrow i$

[3] 計算機 M 上で稼働している言語 P のコンパイラを計算機 M' に、できるだけ簡単な労力で移植したい。このコンパイラは2つのプログラム A, B からなっており、 A は P で書かれたソースプログラムを中間言語 I のプログラムに変換するもので、 I は M のアーキテクチャに依存していない。 B は I で書かれたプログラムを、 M の機械語に変換する。 A, B の P で書かれたソースコードは手元にある。この移植の手順を **T** 図形を用いて説明せよ。

(4ページにかけ) (20点)

[1] ① 左<<>&L

$$\begin{aligned} E &\rightarrow ET' | T \\ T' &\rightarrow +T | -T \\ T &\rightarrow TF' | F \\ F' &\rightarrow \div F | \times F \\ F &\rightarrow i \end{aligned}$$

左再帰除去 G'

$$\begin{aligned} E &\rightarrow TE' \\ E' &\rightarrow T'E' | \epsilon \\ T' &\rightarrow +T | -T \\ T &\rightarrow FF'' \\ F'' &\rightarrow F'F'' | \epsilon \\ F' &\rightarrow \div F | \times F \\ F &\rightarrow i \end{aligned}$$

これは正解

いまだ再帰除去しようとしたらこうなる

$$\begin{aligned} G'' \\ E &\rightarrow TE' \\ E' &\rightarrow +TE' | -TE' | \epsilon \\ T &\rightarrow FT' \\ T' &\rightarrow \div FT' | \times FT' | \epsilon \\ F &\rightarrow i \end{aligned}$$

これは出力は同じだが 予読意とは異なるので 50%正解

② G' : $\text{First}(E) = \text{First}(T) = \text{First}(F) = \{i\}$

$$\text{First}(E') = \{\epsilon, +, -\}$$

$$\text{First}(T') = \{+, -\}$$

$$\text{First}(F') = \{\div, \times\}$$

$$\text{First}(F'') = \{\epsilon, \div, \times\}$$

G'' に対しては

$$\text{First}(E) = \{i\}$$

$$\text{First}(E') = \{+, -, \epsilon\}$$

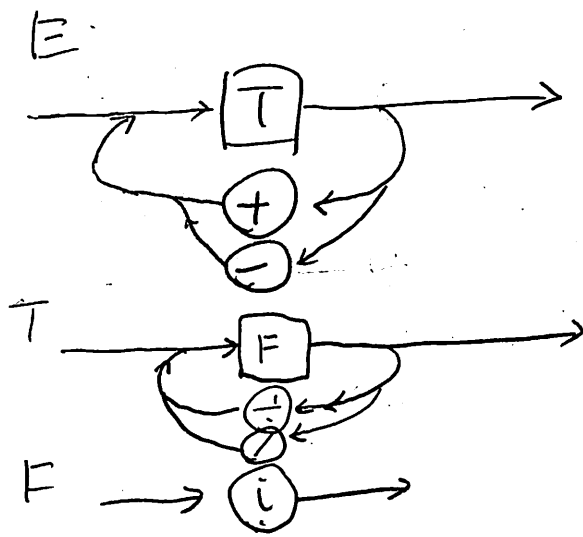
$$\text{First}(T) = \{i\}$$

$$\text{First}(F') = \{\div, \times, \epsilon\}$$

$$\text{First}(F) = \{i\}$$

50%正解

3



[2] ① $\text{follow}(E') = \{\#\}$, $\text{follow}(E) = \{\#\}$

② $I_0: E' \rightarrow \cdot E$
 $E \rightarrow \cdot / E$
 $E \rightarrow \cdot i * E$
 $E \rightarrow \cdot i$

(I_0/E)

$I_2: E \rightarrow i \cdot / E$
 $E \rightarrow i \cdot * E$
 $E \rightarrow i \cdot$

(I_0/i)
 (I_3/i)
 (I_4/i)

$I_3: E \rightarrow i / \cdot E$
 $E \rightarrow \cdot i / E$
 $E \rightarrow \cdot i * E$
 $E \rightarrow \cdot i$

(I_2/i)

$I_4: E \rightarrow i * \cdot E$
 $E \rightarrow \cdot i / E$
 $E \rightarrow \cdot i * E$
 $E \rightarrow \cdot i$

$(I_2/*)$

$I_5: E \rightarrow i / E \cdot$

(I_3/E)

$I_6: E \rightarrow i * E \cdot$

(I_4/E)

	i	/	*	E'	E
0	2				1
1					
2		3	4		
3	2				5
4	2				6
5					
6					

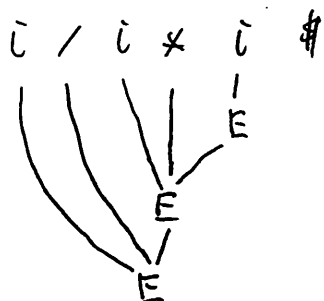
③

state	Action				goto	
	i	/	*	#	E'	E
0	S2					1
1				Accept (R ϕ)		
2		S3	S4	R3		
3	S2					5
4	S2					6
5				R1		
6				R2		

④

stack	input
ϕ	i / i * i #
$\phi i 2$	/ i * i #
$\phi i 2 / 3$	i * i #
$\phi i 2 / 3 i 2$	* i #
$\phi i 2 / 3 i 2 * 4$	i #
$\phi i 2 / 3 i 2 * 4 i 2$	#
$\phi i 2 / 3 i 2 * 4 E 6$	#
$\phi i 2 / 3 E 5$	#
$\phi E 1$	#
Accept	

⑤



上下逆 OK.

[3] 前提として

AのY-2



BのY-2



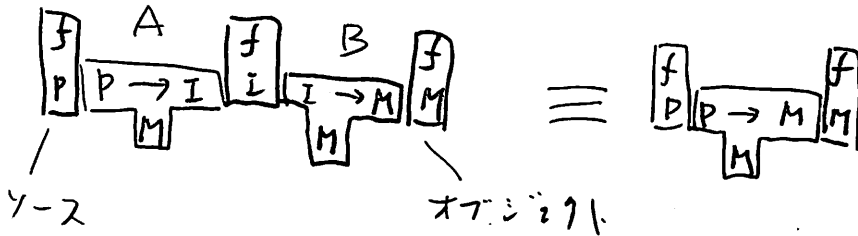
が利用できる (のとする)

元のコンパイラ

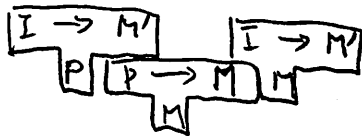


は 2 段ステップ、P になる、とあり

これらのY-2をコンパイルして連続して動かすことで実現したい

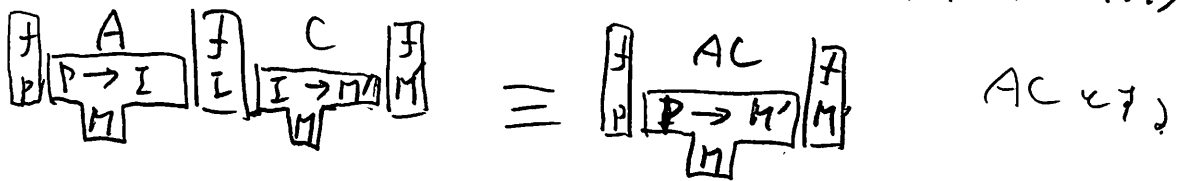


手で書く部分を簡単にするためには、CのY-2 I → M P を手で書く
これを M 上でコンパイルすると

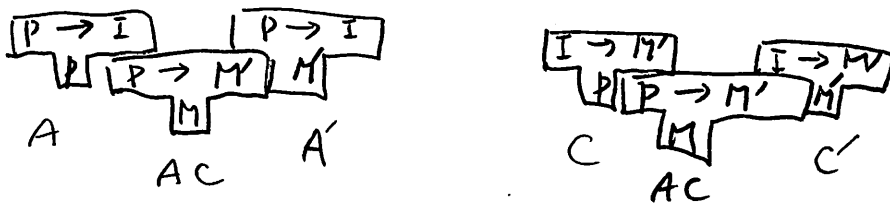


C
I → M
M が得られた

A と C を組合せると M 上での P → M' への Y-2 コンパイルが得られた



A, C だけでなくの Y-2 を AC でコンパイルする (できるものを
それぞれ A', C' とする)



A', C' を組合せて 目的のコンパイルをする (D)

