

問1 下の文法について、次の(1)~(5)の問いに答えなさい。

$$E \rightarrow E + T \mid T$$

$$T \rightarrow T * F \mid F$$

$$F \rightarrow 1 \mid (E) \quad \text{※}E \text{ などの大文字は構文変数}$$

- (1) この文法のもとで式 $1 * (1 + 1)$ をトップダウン構文解析, ボトムアップ構文解析する. 途中の $1 * ($ まで解析したときに作成される構文解析木の部分を, それぞれ示しなさい.
- (2) 2行目の「 $T \rightarrow T * F \mid F$ 」を, $\llbracket \quad \rrbracket$ (0回以上の繰返し) を用いた式に書き換えなさい.
- (3) 上の(2)によって, もとの「 $T \rightarrow T * F \mid F$ 」の持つ, トップダウン構文解析を妨げる性質が除去される. その性質とは何か.
- (4) この文法における初期構文変数を示しなさい.
- (5) 次の(a)~(d)に対する構文解析木を示しなさい.

(a) $1 * 1$ (b) $1 + 1 + 1$ (c) (1) (d) $1 + 1 * (1 + 1)$

問2 次の文章を読んで、(1)~(7)の問いに答えなさい。

下向き構文解析では、次の入力字句をヒントとして $A \rightarrow B \mid C$ において A を B または C のいずれに書き換えるべきかを決定する。

次の字句が (①) に含まれるなら B を選択すればよい。同様に次の字句が (②) に含まれるなら C を選択する。

ただし、 B に ε が含まれる場合は、次の字句が (③) に含まれる場合も、 B を選択すべきである。 B に ε が含まれる場合も考慮して、 B を選択する基準になる集合を (④) で表す。次の記号がこの集合に含まれるなら、 B を選択すればよい。

- (1) ①~④を埋める適切な語句や式などを答えなさい。
- (2) $\text{FIRST}(1)$ を求めなさい (1 は終端記号)。
- (3) 「 $F \rightarrow 1 \mid (E)$ 」という規則によって $\text{FIRST}(F)$ に加えられる記号を求めなさい。
- (4) 上の(3)によって求められた $\text{FIRST}(F)$ を利用して、「 $T \rightarrow T * F \mid F$ 」という規則によって $\text{FIRST}(T)$ に加えられる記号を求めなさい。
- (5) 「 $T \rightarrow T * F \mid F$ 」という規則があるときに $\text{FOLLOW}(T)$ に加えられる記号を求めなさい。
- (6) B が ε を含まない場合、含む場合のそれぞれの $\text{DIRECTOR}(B)$ 集合を、 $\text{FIRST}(B)$ と $\text{FOLLOW}(B)$ を用いて表しなさい。
- (7) どのような場合に $A \rightarrow B \mid C$ を含む文法が $\text{LL}(1)$ でなくなるか。 DIRECTOR 集合を用いて説明しなさい。

J4 言語処理系 2013 年度前期期末試験問題(2/2)

2013/09/17

問 3 下は授業中の演習で扱った電卓構文解析プログラムの一部である。①～⑥に入る適切な式等を補い、プログラムを完成させなさい。ただし終端記号とシンボルの対応は、下の表のとおりとする。

```

/* E ::= [ + | - ] T { ( + | - ) T } */
int parse_expression(void){
    if ( nextsym.sym == ① || nextsym.sym == ② ){
        nextsym = scanner_get_next_sym();
    }
    parse_term();
    while(nextsym.sym == SYM_PLUS || nextsym.sym == SYM_MINUS){
        nextsym = scanner_get_next_sym();
        ③
    }
}

/* T ::= F { ( * | / ) F }
int parse_term(void){ /* 省略 */ }

/* F ::= N | '(' E ')' */ ※ '(' , ')' は終端機能の ( , )
int parse_factor(void){
    if (nextsym.sym == ④ ){
        nextsym = scanner_get_next_sym();
    }else if (nextsym.sym == SYM_LPAREN){
        nextsym = scanner_get_next_sym();
        ⑤
        if (nextsym.sym != ⑥ ){
            ERROR("'" が必要です\n");
        }
        nextsym = scanner_get_next_sym();
    }else{
        ERROR("<N> または '(' が必要です");
    }
}

```

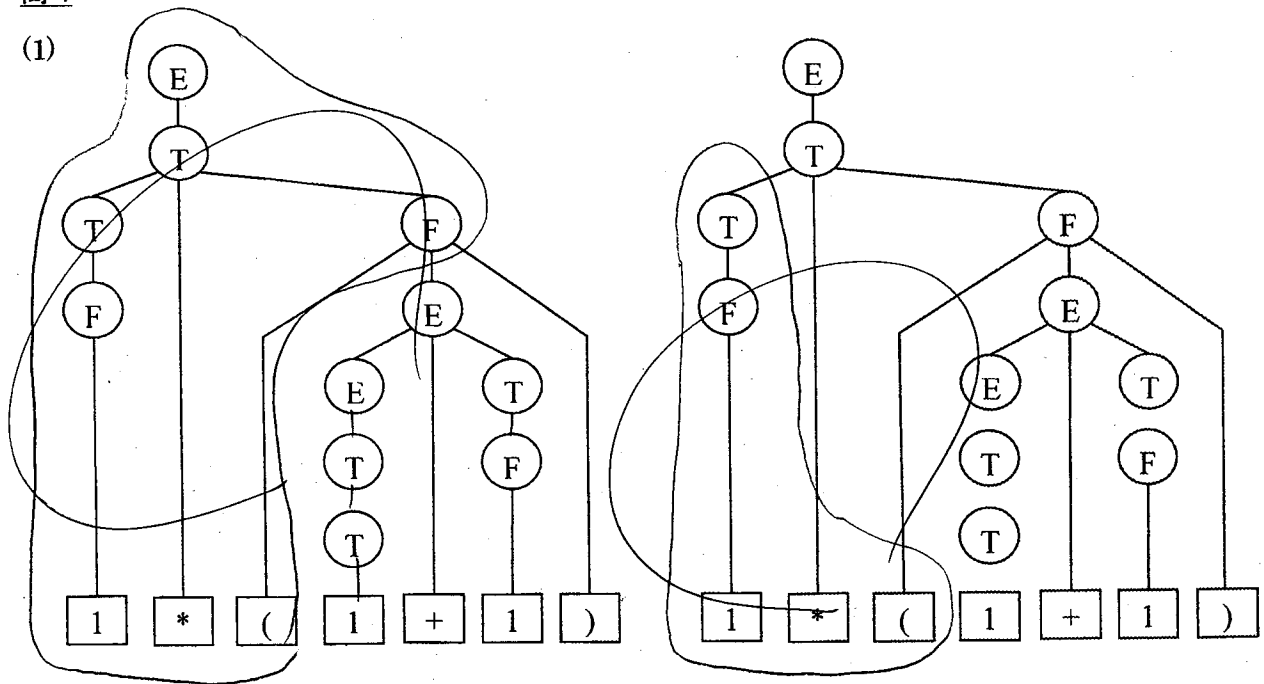
終端記号とシンボル対応表

終端記号	シンボル	終端記号	シンボル
+	SYM_PLUS	(SYM_LPAREN
-	SYM_MINUS)	SYM_RPAREN
*	SYM_ASTERISK	数値(N)	SYM_CONSTANT_INT
/	SYM_SLASH		

学籍番号 J_

問 1

(1)



トップダウン構文解析

ボトムアップ構文解析

(2)

$T \rightarrow F * F$

(3)

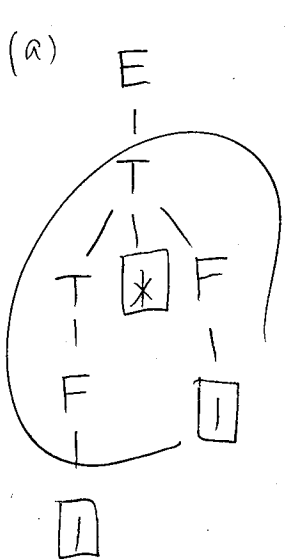
左再帰性

(4)

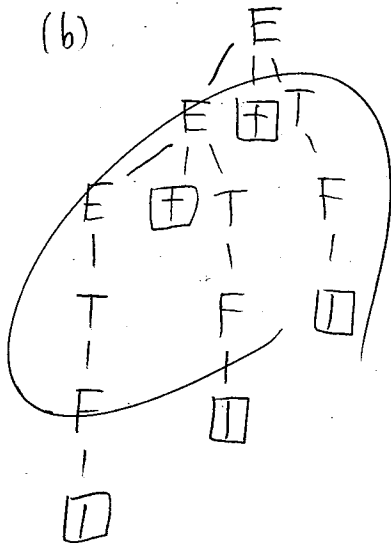
$E \rightarrow 1$

(5) (スペースが足りないときは裏面を使うこと)

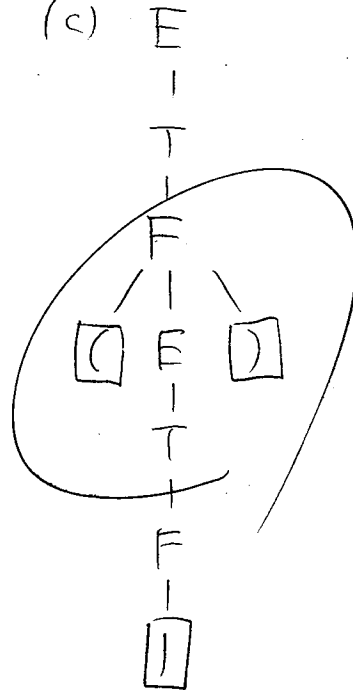
(a)



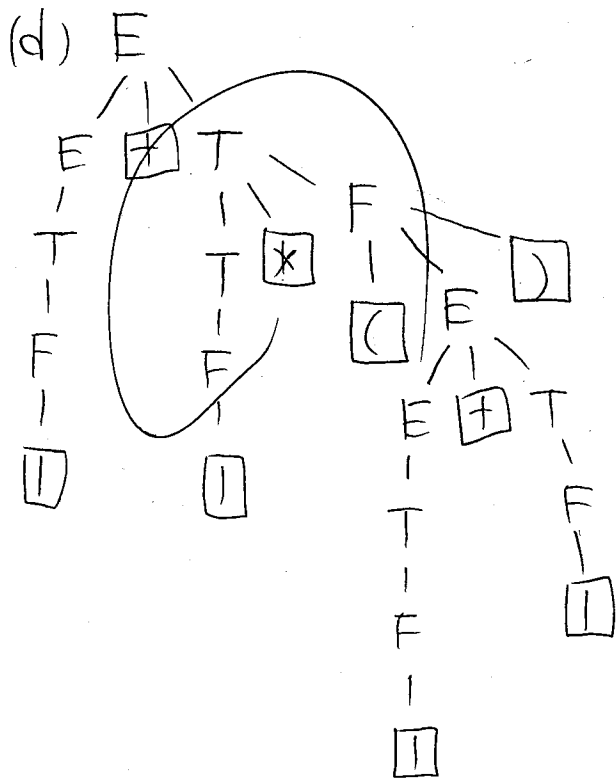
(b)



(c)



(d)



学籍番号

問2

(1) ① FIRST(B)② FIRST(C)③ FOLLOW(B)④ DIRECTOR(B)(2) A(3) B(4) X, C(5) *(6) 含まない場合 DIRECTOR(B) = FIRST(B)

含む場合

DIRECTOR(B) =FIRST(B) \cup FOLLOW(B)(7) $DIRECTOR(B) \cap DIRECTOR(C) \neq \emptyset$ であるとき。

問3

① SYM-PLUS② SYM-MINUS③ parse_term();④ SYM-CONSTANT-INT⑤ parse_expression();⑥ SYM-RPAREN