

1. 言語解析方式

オートマトン、演算子順位構文解析(operator precedence parsing)、再帰的下向き構文解析(recursive decent parsing)の3つについて、それぞれ4～5行で、その概要と長所、短所、適する用途を書きなさい。

2. 正規表現

(1) 正規表現 $(a b \mid c)^* c (b \mid a)^*$ に対する非決定性有限オートマトンを書きなさい。ここで、 $\mid$ は「または」を表し、 $*$ は0回以上の繰り返しを表すものとする。したがって、 $(a b \mid c)^*$ は $a b$ または c の文字列の0回以上の繰り返しを表す。

(2) 上記オートマトンを決定性有限オートマトンに変換しなさい。

3. 演算子順位法

算術式

$$(b - c) * (d + e) + a$$

をポーランド記法(正確には逆ポーランド記法)に変換しなさい。

4. 下向き構文解析

C言語の if 文

if (条件式) 文 else 文

を次ぎのように単純化したとする。

条件式は2つの算術式を比較演算子($=$, $!=$, $<$, $>$)で結合したものである。

文は代入文 変数 = 算術式 ; に限定する。変数は識別子に限定する。

算術式は、変数と定数、あるいはそれらに対する加減乗除式と、それらを括弧でくくった(算術式)に限定する。

識別子と定数は終端記号として扱い、文字にまでは分解しない。

- (1) この if 文の構文規則を下向き構文解析向きにBNFで記述しなさい。
- (2) その各非終端記号に対する First 集合と Follow 集合を求めなさい。
- (3) それに対応する下向き構文解析表を作りなさい。
- (4) 各非終端記号に対する再帰的下向き構文解析の処理概要を書きなさい。
(教科書 63 ページ、あるいは講義で説明したような形式)

5. コード生成

Tiny C の文

```
if (a > b) {  
    if (a < c)  
        x = a - b;  
    else  
        x = c - b;  
}
```

に対して生成すべきコード列を教科書 123 ページ(プリント Tiny C コンパイラの概要) 9 ページ) に示したコードで書きなさい。