

コンパイラ（'10年度）・期末テスト問題用紙

（'10年7月29日（木）・16:20～17:50）

解答上、その他の注意事項

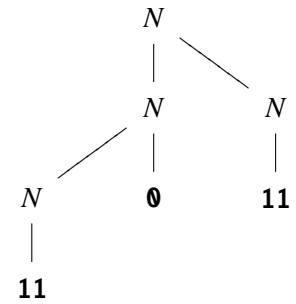
- I. 問題は、問 I～VI までである。
- II. 解答用紙の右上の欄に学籍番号・名前を記入すること。
- III. 解答欄を間違えないよう注意すること。
- IV. 解答中の文字（特に a と d）がはっきりと区別できるよう注意すること。
- V. 持ち込みは不可である。筆記用具・時計・学生証以外のものは、かばんの中などにしまうこと。
- VI. テストの配点は 80 点である。合格はレポートの得点を加えて、100 点満点中 60 点以上とする。

I. (Backus-Naur 記法)

次のような BNF で表される文法を考える。

$$\begin{array}{l}
 N \rightarrow N 0 \\
 \quad | \quad N N \\
 \quad | \quad 11 \\
 \quad | \quad 1001
 \end{array}$$

例: 11011 に対する解析木



次の各記号列について、上の BNF の非終端記号 N から導出されるものには、その解析木 (parse tree) を右の例にならって書き、導出されないものには X を記せ。(解析木は一通りとは限らないが、そのうちひとつを書けば良い。)

- (1) 1100100
- (2) 1110010
- (3) 1001110

II. (正規表現)

以下の文字列について、「 $(ba|a(bc)^*)^*$ 」という正規表現にマッチする先頭からの最長の部分文字列の文字数を答えよ。例えば、baabcbcbcbcab という文字列について考えると、その先頭の部分文字列 baabcbc は上の正規表現にマッチするが、それより長い先頭からの部分文字列: baabcbcb, baabcbcbb, ..., baabcbcbcbcab はいずれもマッチしないので、マッチする先頭からの最長の部分文字列の文字数は 7 となる。(なお、文字列の途中からの部分文字列は考えなくて良い。)

- (1) abcbcbbaabcbba
- (2) bacabcbbaabca
- (3) abcbcbabcbab
- (4) abcbbaabcbbaac

III. (コンパイラのフェーズ)

コンパイラは、字句(単語)を切り分ける字句解析フェーズ、プログラムの構造を木の形に表す構文解析フェーズ、変数の宣言や型のチェックを行なう意味解析(静的解析)フェーズ、目的のコードを生成するコード生成フェーズなどに概念的に分けることができる。

次の(1)~(3)のC言語のプログラムにはそれぞれ誤りがある。コンパイラのどのフェーズで最初に誤りが検出されるか?(あるいはされないか?)もっとも適当なものを下の選択肢(A)~(E)から選べ。なお、(1)~(3)のいずれも単独でコンパイルされ、標準ライブラリとのみリンクされるものとする。(つまり、他のファイルに変数や関数が定義されていることはない。)

- (1) (ブロックの“{”~“}”の代わりに普通の括弧“(”~“)”を使った。)

```
#include <stdio.h>

int main(void) (
    int i;
    for (i=0; i<10; i++) (
        printf("Hello World!¥n");
    )
)
```

- (2) (正弦関数 double sin(double) に文字列を渡した。)

```
#include <stdio.h>
#include <math.h>

int main(void) {
    printf("sin(45 °) = %f¥n", sin("45 °"));
    return 0;
}
```

- (3) (文字列リテラルの終わりを示す「\n」を忘れた。)

```
#include <stdio.h>

int main(void) {
    printf("Hello! World¥n");
    return 0;
}
```

(1)~(3)の選択肢

- (A) 字句解析フェーズでエラーが検出される。
- (B) 構文解析フェーズでエラーが検出される。
- (C) 意味解析フェーズでエラーが検出される。
- (D) コード生成フェーズでエラーが検出される。
- (E) 実行時にエラーとなるか、まったくエラーにならない(が作成者の意図と異なる動作をする)。

IV. (演算子順位法)

次のBNFで表される文法を演算子順位法により構文解析する。

$$E \rightarrow \text{id} \mid E \text{ "!!" } E \mid E \text{ ":" } E \mid E \text{ "==" } E \mid \text{ "(" } E \text{ ")"}$$

ただし、idはアルファベット1文字からなるトークンを表す。

この文法は曖昧なので、優先順位と結合性について次のように決めておく。

「!!」は左結合、「:」は右結合、「==」は非結合であり、「!!」は「:」よりも優先順位が高く、「:」は「==」よりも優先順位が高いものとする。

つまり、下表中の左の欄の式は、右の欄の式として解釈される。

式	解釈
a !! b !! c	(a !! b) !! c
a : b : c	a : (b : c)
a == b == c	構文エラー
a !! b : c	(a !! b) : c
a : b !! c	a : (b !! c)
a !! b == c	(a !! b) == c
a == b !! c	a == (b !! c)
a : b == c	(a : b) == c
a == b : c	a == (b : c)

以下の演算子順位行列の空欄(1)~(4)を <(シフト) >(還元) X(エラー)のうちもっとも適切なもので埋めよ。

左 \ 右	!!	:	==	()	id	終
始	<	<	<	<	X	<	≠
!!	(1)	>	(2)	<	>	<	>
:	<	(3)	>	<	>	<	>
==	<	<	(4)	<	>	<	>
(<	<	<	<	≠	<	X
)	>	>	>	X	>	X	>
id	>	>	>	X	>	X	>

V. (再帰下降構文解析)

次のようなBNFで定義された文法に対して再帰下降構文解析ルーチンを作成する。

$$S \rightarrow S ";" S \mid \mathbf{id} " := " E \mid \mathbf{pr} "(" E ")"$$

$$E \rightarrow \mathbf{id} \mid E "+" E \mid "(" S "@" E ")"$$

ただし、「 S 」、「 E 」は非終端記号で、「 $;$ 」、「 \mathbf{id} 」、「 $:=$ 」、「 \mathbf{pr} 」、「 $($ 」、「 $)$ 」、「 $+$ 」、「 $@$ 」は終端記号とする。開始記号 (start symbol) は S である。

(1) E から左再帰を除去すると、次のようなBNFが得られる。

$$E \rightarrow \mathbf{id} E' \mid "(" S "@" E ")" E'$$

$$E' \rightarrow \varepsilon \mid "+" E E'$$

これを参考にして、 S から左再帰を除去せよ。補助的に導入する非終端記号は S' とせよ。

以下の(2)~(4)は、(1)で S と E から左再帰を除去して得られたBNFについて答えよ。

- (2) $First(S')$ を求めよ。
- (3) $Follow(S')$ を求めよ。
- (4) 下の構文解析表の E の行を埋めよ。

	$;$	\mathbf{id}	$:=$	\mathbf{pr}	$($	$)$	$+$	$@$	$\$$
$S \rightarrow$									
$S' \rightarrow$									
$E \rightarrow$									
$E' \rightarrow$									

(4)の解答は次の選択肢から選べ。

- (A). $\mathbf{id} E'$
- (B). $(" S "@" E ")" E'$
- (C). ε
- (D). $"+" E E'$
- (E). \times

ただし、 \times は“構文誤り”を示す。

VI. (LR 構文解析)

次のような文法 (… の後の I, II などは生成規則の番号)

$$\begin{aligned}
 E &\rightarrow \mathbf{id} && \dots I \\
 &| \{ X \} && \dots II \\
 &| E \{ X \} && \dots III \\
 &| E \# \mathbf{id} && \dots IV \\
 X &\rightarrow \mathbf{id} = E && \dots V \\
 &| X ; \mathbf{id} = E && \dots VI
 \end{aligned}$$

に対して、LR 構文解析表を作成する。ただし、

- 「 E 」, 「 X 」は非終端記号で、「 \mathbf{id} 」, 「 $\{$ 」, 「 $\}$ 」, 「 $\#$ 」, 「 $=$ 」, 「 $;$ 」は終端記号とする。このうち、「 \mathbf{id} 」はアルファベット 1 文字からなるトークンを表す。
- 開始記号 (start symbol) は E である。

bison の出力する LR 構文解析表は次のようになる。(注: bison に $-v$ オプションを指定することによって、LR 構文解析表をファイルに出力させることができる。)

	id	{	}	#	=	;	\$	E	X
①	shift ①	shift ②						goto ③	
①	reduce I								
②	shift ④								goto ⑤
③		shift ⑦		shift ⑧			shift ⑥		
④					shift ⑨				
⑤			shift ⑩			shift ⑪			
⑥	accept								
⑦	shift ④								goto ⑫
⑧	shift ⑬								
⑨	shift ①	shift ②						goto ⑭	
⑩	reduce II								
⑪	shift ⑮								
⑫			shift ⑯			shift ⑰			
⑬	reduce IV								
⑭	reduce V	shift ⑦	reduce V	shift ⑧	reduce V				
⑮					shift ⑰				
⑯	reduce III								
⑰	shift ①	shift ②						goto ⑱	
⑱	reduce VI	shift ⑦	reduce VI	shift ⑧	reduce VI				

注: shift ⑮は、「シフトして状態⑮へ遷移」、goto ⑮は、「状態⑮へ遷移」、reduce X は、「生成規則 X を使って還元」を表す。

次の入力に対して、↑の次(右)の記号をシフトした直後の(つまりシフトしたあと、還元がまだ起こっていないときの)スタックの状態はどのようになっているか?

(1) $a\{b=c;s=t;x=y\}$

(2) $a\{b=c\}\{s=t\}\{x=y\}$

下の選択肢(1)~(2)共通)から選べ。(左がスタックの底とする)

- (A). $\textcircled{0}E\textcircled{3}\{\textcircled{7}id\textcircled{4}\}$
- (B). $\textcircled{0}E\textcircled{3}\{\textcircled{7}X\textcircled{12};\textcircled{1}id\}$
- (C). $\textcircled{0}E\textcircled{3}\{\textcircled{7}id\textcircled{4}=\textcircled{9}E\textcircled{14};\textcircled{1}id\}$
- (D). $\textcircled{0}E\textcircled{3}\{\textcircled{7}id\textcircled{4}=\textcircled{9}E\textcircled{14}\}\textcircled{16}\{\textcircled{7}id\textcircled{4}\}$
- (E). $\textcircled{0}E\textcircled{3}\{\textcircled{7}X\textcircled{12};\textcircled{1}id\textcircled{4}=\textcircled{9}E\textcircled{14};\textcircled{1}id\}$
- (F). $\textcircled{0}E\textcircled{3}\{\textcircled{7}id\textcircled{4}=\textcircled{9}E\textcircled{14};\textcircled{1}id\textcircled{4}=\textcircled{9}E\textcircled{14};\textcircled{1}id\}$
- (G). $\textcircled{0}E\textcircled{3}\{\textcircled{7}id\textcircled{4}=\textcircled{9}E\textcircled{14}\}\textcircled{16}\{\textcircled{7}id\textcircled{4}=\textcircled{9}E\textcircled{14}\}\textcircled{16}\{\textcircled{7}id\textcircled{4}\}$

コンパイラ ('10年度)・ 期末テスト 解答用紙 ('10年 7月 29日)

学籍番号		氏名	
------	--	----	--

I. (Backus-Naur 記法) (4×3)

(1).	(2).	(3).

II. (正規表現) (4×4)

(1).		(2).		(3).		(4).	
------	--	------	--	------	--	------	--

III. (コンパイラのフェーズ) (4×3)

(1).		(2).		(3).	
------	--	------	--	------	--

IV. (演算子順位法) (4×4)

(1).		(2).		(3).		(4).	
------	--	------	--	------	--	------	--

裏ページに続く。

