

今回の内容

5.1 プログラミング言語 Minimum C . . . . .	5-1
5.2 演習問題 . . . . .	5-3

## 5.1 プログラミング言語 Minimum C

今回以降、例題として **Minimum C** という C に似たプログラミング言語を考え、これをソース言語と想定して具体的なコンパイラを構築していくことにします。次の2つは Minimum C プログラムの例です<sup>1</sup>。

```

pow3.mc
1  int x, y, z;
2
3  pow ()
4  {
5      z = 1;
6      while (y > 0) {
7          z = z * x;
8          y = y - 1;
9      }
10 }
11
12 main ()
13 {
14     input x;
15     input y;
16     pow();
17     print z;
18 }

```

```

gcd.mc
1  gcd (x, y)
2  {
3      int z;
4
5      if (x < y) {
6          z = x;
7          x = y;
8          y = z;
9      }
10
11     while (y != 0) {
12         z = y;
13         y = x % y;
14         x = z;
15     }
16
17     return x;
18 }
19
20 main ()
21 {
22     int a, b;
23
24     input a;
25     input b;
26     print gcd(a, b);
27 }

```

C と比較すると、Minimum C には以下のような制限があります。

1. データ型は `int` 型のみで、演算子の種類が非常に限られている。
2. 入出力のための特別な構文 (`input` 文と `print` 文) が用意されている。
3. `for` 文や `switch` 文、`do` 文、`break` 文、`continue` 文、`goto` 文はない。
4. `#define` や `#include` などの前処理指令 (プリプロセッサ制御文) は使用できない。

<sup>1</sup>ただし行頭の数には行番号で、プログラムの一部ではありません

## Minimum C の文法

Minimum C は C のサブセットであり、その文法を BNF 記法で記述すると以下のようになります。〈*Prog*〉が Minimum C で書かれたプログラム全体を表す非終端記号です。

$$\begin{aligned}\langle Prog \rangle &::= \{ \langle VarDef \rangle \mid \langle FunDef \rangle \} . \\ \langle VarDef \rangle &::= \text{"int"} \langle Ident \rangle \{ \text{","} \langle Ident \rangle \} \text{";" } . \\ \langle FunDef \rangle &::= \langle Ident \rangle \text{"("} [ \langle Ident \rangle \{ \text{","} \langle Ident \rangle \} ] \text{")" } \langle Block \rangle . \\ \langle Block \rangle &::= \text{"{"} [ \langle VarDef \rangle ] \{ \langle Statement \rangle \} \text{"}" } . \\ \langle Statement \rangle &::= \langle Ident \rangle \text{"="} \langle Exp \rangle \text{";" } \mid \langle Ident \rangle \langle FunArgs \rangle \text{";" } \\ &\mid \text{"return"} \langle Exp \rangle \text{";" } \mid \text{"print"} \langle Exp \rangle \text{";" } \mid \text{"input"} \langle Ident \rangle \text{";" } \\ &\mid \text{"if"} \text{"("} \langle Cond \rangle \text{")" } \langle Statement \rangle [ \text{"else"} \langle Statement \rangle ] \\ &\mid \text{"while"} \text{"("} \langle Cond \rangle \text{")" } \langle Statement \rangle \\ &\mid \langle Block \rangle . \\ \langle FunArgs \rangle &::= \text{"("} [ \langle Exp \rangle \{ \text{","} \langle Exp \rangle \} ] \text{")" } . \\ \langle Factor \rangle &::= \langle Ident \rangle [ \langle FunArgs \rangle ] \mid \langle Number \rangle \mid \text{"("} \langle Exp \rangle \text{")" } . \\ \langle MultDiv \rangle &::= \text{"*"} \mid \text{"/" } \mid \text{"\%"} . \\ \langle Term \rangle &::= \langle Factor \rangle \{ \langle MultDiv \rangle \langle Factor \rangle \} . \\ \langle PlusMinus \rangle &::= \text{"+" } \mid \text{"-"} . \\ \langle Exp \rangle &::= [ \langle PlusMinus \rangle ] \langle Term \rangle \{ \langle PlusMinus \rangle \langle Term \rangle \} . \\ \langle Rel \rangle &::= \text{"==" } \mid \text{"!=" } \mid \text{"<" } \mid \text{">" } \mid \text{"<=" } \mid \text{">=" } . \\ \langle Cond \rangle &::= \langle Exp \rangle \langle Rel \rangle \langle Exp \rangle .\end{aligned}$$

空文は許されないこと、変数への代入は文であり、式ではないこと、if 文の構文には曖昧性があること、if 文や while 文の条件部には2つの式の比較しか書くことができないことに注意してください。また、BNF 記法では表現されていない文法上の制限として以下のようなものがあります。

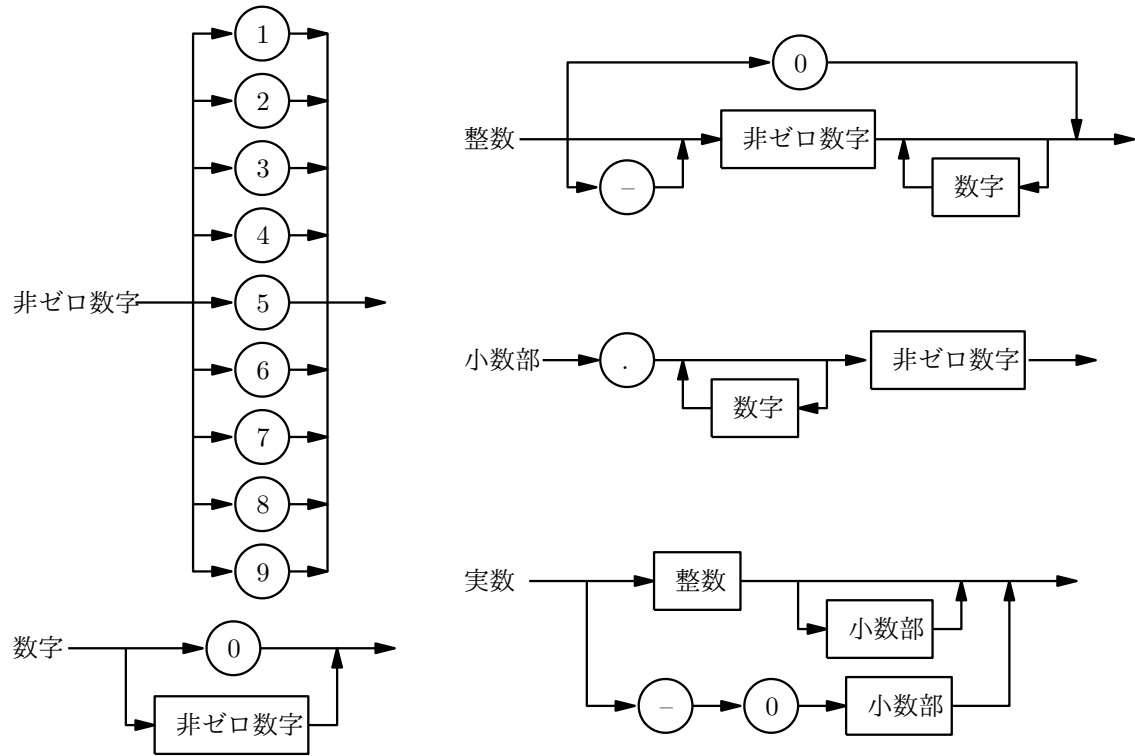
1. 〈*Number*〉は10進数表記の非負整数を表す。
2. 〈*Ident*〉は変数名や関数名などの識別子であり、“int”、“return”、“while”などの予約語とは重ならないものとする。
3. 〈*Number*〉や〈*Ident*〉は字句解析部がトークンとして切り出す。この際、トークン間の空白類は無視される。
4. Cと同様に、〈*VarDef*〉によってあらかじめ宣言されていない変数を使うことはできない。ただし、関数は後から定義してもよい。
5. Cと同様に、main という名前の関数が定義されていなければならない。プログラム全体の実行は、この main から開始される。

## 5.2 演習問題

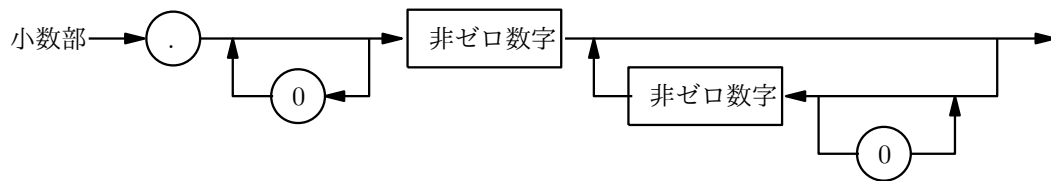
1.  $x$  と  $f$  が  $\langle Ident \rangle$  であり、 $123$  が  $\langle Number \rangle$  であるとして、 $x = f(-123);$  の  $\langle Statement \rangle$  に対する構文木を書きなさい。
2. Minimum C の文法の `if` 文に関する曖昧性を示しなさい。
3. BNF 記法で書かれた Minimum C の文法に従って、各非終端記号の構文図を書きなさい。また、できあがった構文図が決定的でないことを示しなさい。

付録：第3回演習問題の解答例

1.

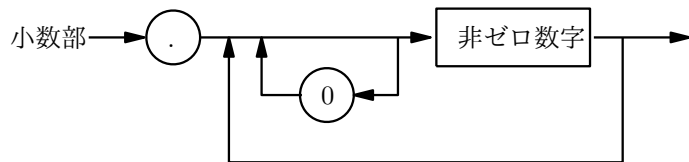


2.

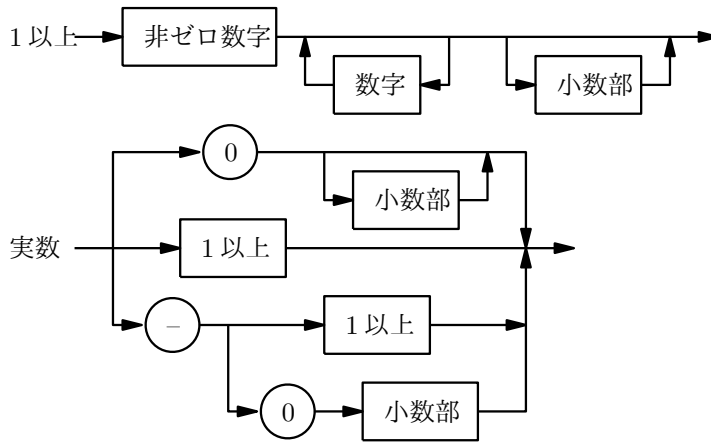


$\langle \text{小数部} \rangle ::= \text{"."} \{ \text{"0"} \} \langle \text{非ゼロ数字} \rangle \{ \{ \text{"0"} \} \langle \text{非ゼロ数字} \rangle \}.$

(構文図の別解)



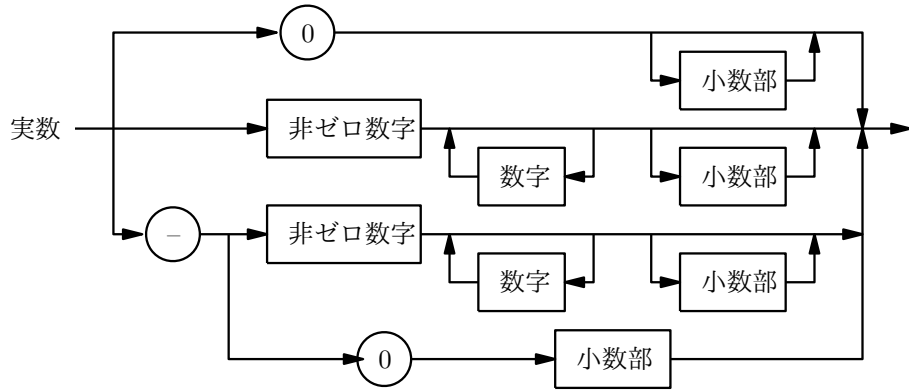
3.



$\langle 1以上 \rangle ::= \langle \text{非ゼロ数字} \rangle \{ \langle \text{数字} \rangle \} [ \langle \text{小数部} \rangle ] .$

$\langle \text{実数} \rangle ::= \text{"0"} [ \langle \text{小数部} \rangle ] \mid \langle 1以上 \rangle \mid \text{"-"} ( \langle 1以上 \rangle \mid \text{"0"} \langle \text{小数部} \rangle ) .$

(別解)



$\langle \text{実数} \rangle ::= \text{"0"} [ \langle \text{小数部} \rangle ] \mid \langle \text{非ゼロ数字} \rangle \{ \langle \text{数字} \rangle \} [ \langle \text{小数部} \rangle ]$

$\mid \text{"-"} ( \langle \text{非ゼロ数字} \rangle \{ \langle \text{数字} \rangle \} [ \langle \text{小数部} \rangle ] \mid \text{"0"} \langle \text{小数部} \rangle ) .$

(構文図の別解)

