
Tiny C言語仕様

はじめに

- 本資料は、C言語のサブセットであるプログラミング言語の言語仕様を記したものである
- 当該言語をTiny C言語と呼ぶ

表記方法

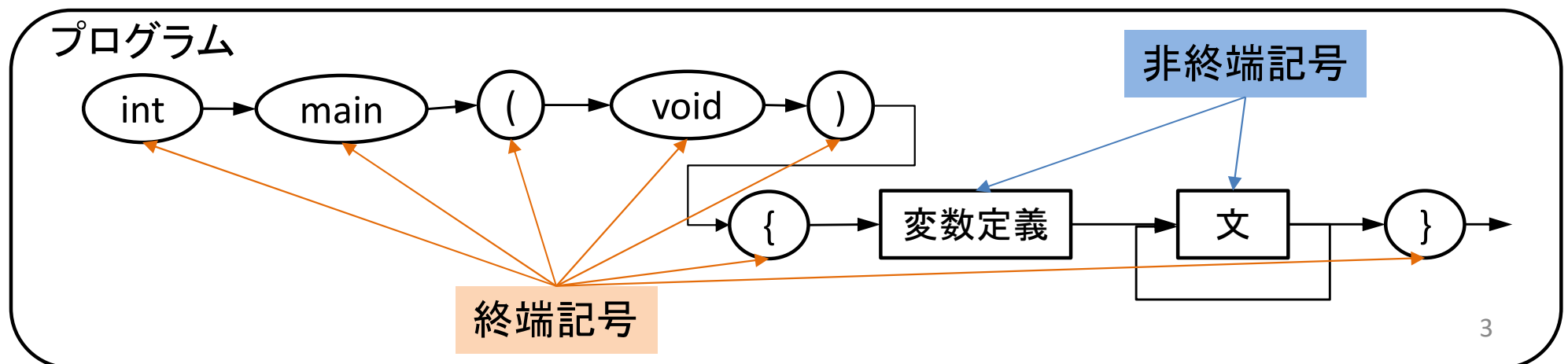
- 本資料で使用する言語仕様は構文則を用いる
- 本資料の固有の記述方法を下記に示す
 - 構文は終端記号と非終端記号からなる
 - 終端記号は文字列であり、楕円で記す
 - 非終端記号は他で定義しており、長方形で記す
 - なお、コメントを記すことができる
 - コメントは”//”で始まり、行末までとする
 - 例

– // This is comments.

←一行全体がコメント

– int main() { // “int main() {“ isn't comment

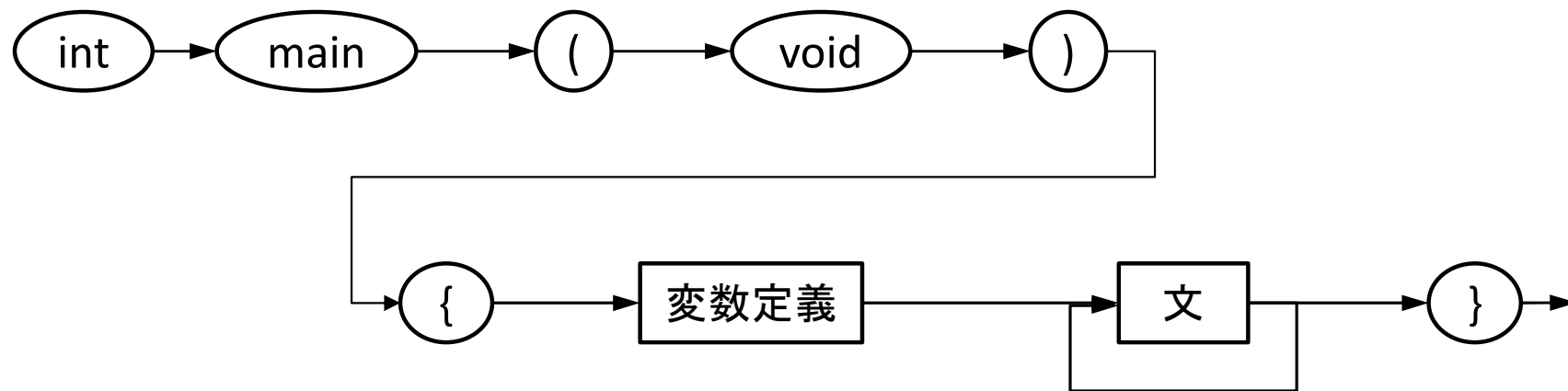
←行の途中からコメント



Tiny Cの構文式 ~プログラム

- 当該言語は、main関数のみからなる
 - 関数をユーザ定義することはできない
 - main関数はint型である
 - main()関数は引数を取らない
 - main()関数は変数定義と文からなる
 - 変数定義は1回のみであるが、文は複数を含むことができる

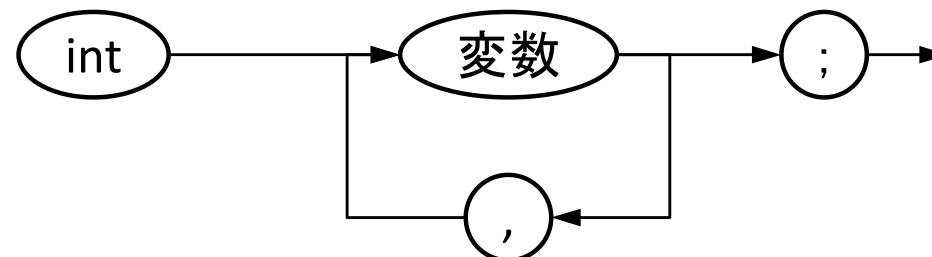
プログラム



Tiny Cの構文式 ～変数定義

- 変数の型は整数型(int)のみとする
 - 変数定義を行う(宣言)として、“int ”を記述する
 - 変数は、0個以上を定義することができる
 - 同じ名前の変数を定義することはできない
- 変数名の制約は下記の通り
 - 変数は先頭が英字で、二文字目以降は英数字とする
 - 変数名に使用する英字は大小を許容する
 - 変数名は16文字以内とする(16文字の変数名は非許容)

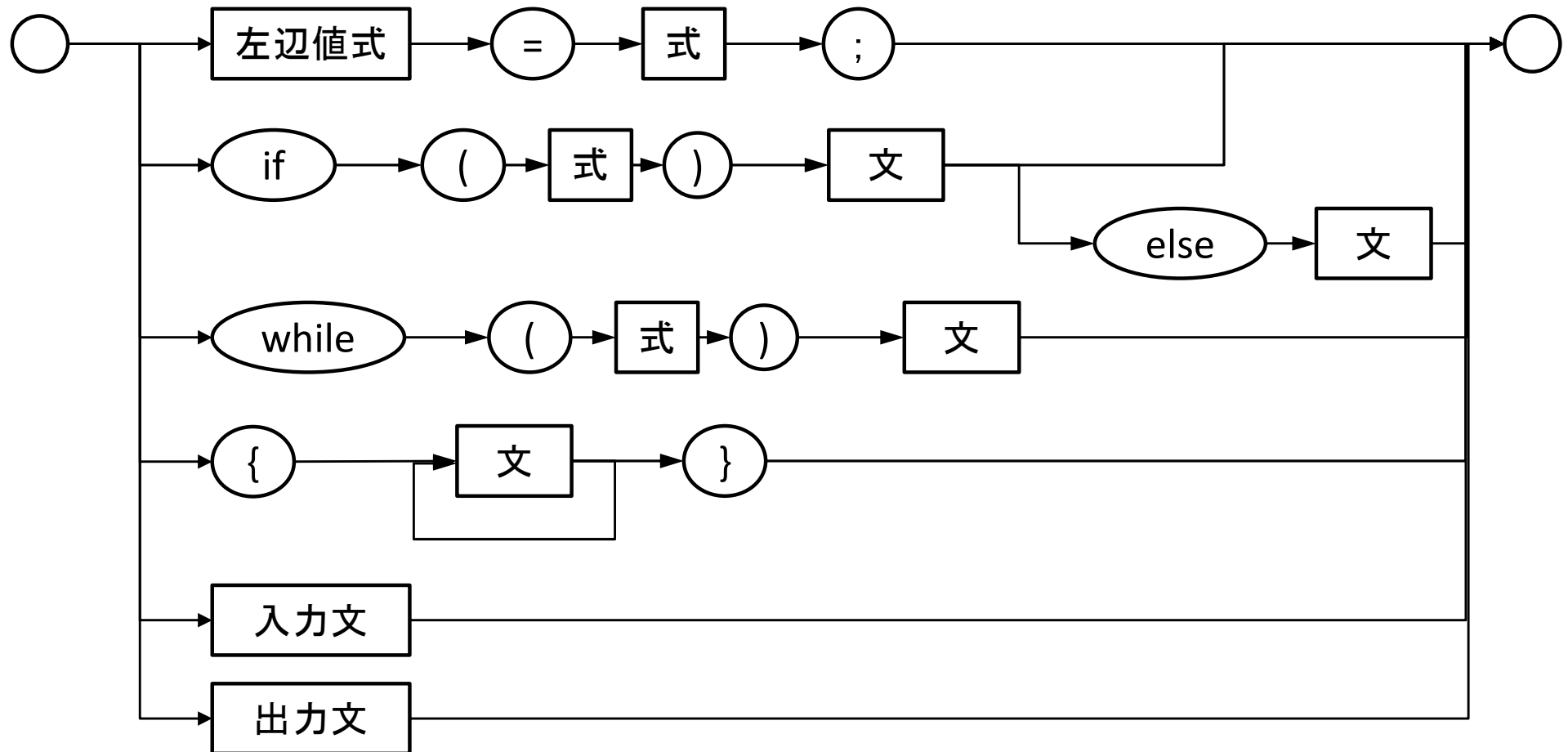
変数定義



Tiny Cの構文式 ～文

- 文は、代入文・if文・while文・複文・入出力文からなる

文



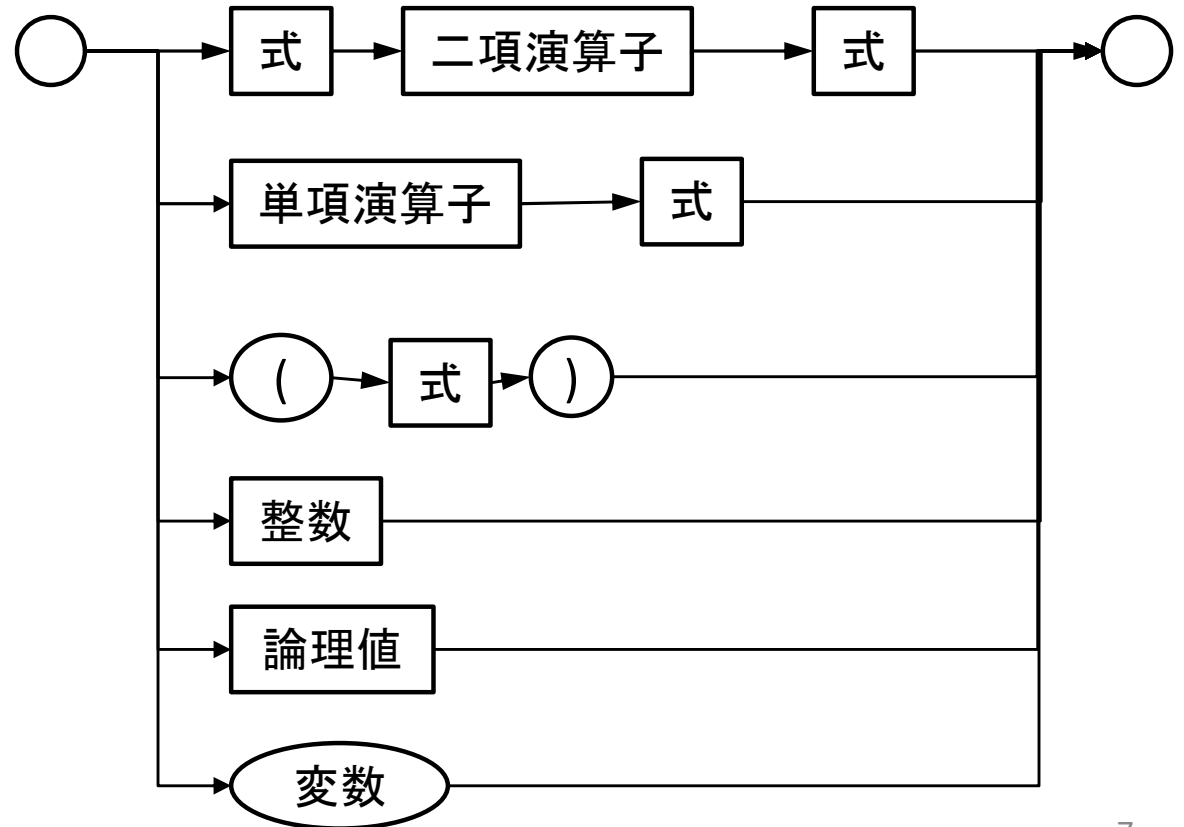
Tiny Cの構文式 ～式

- 式には、左辺値式と式の二種類がある
- 左辺値式では代入文の左辺を定義している
- 式は、代入文をはじめとする式を定義している

左辺値式



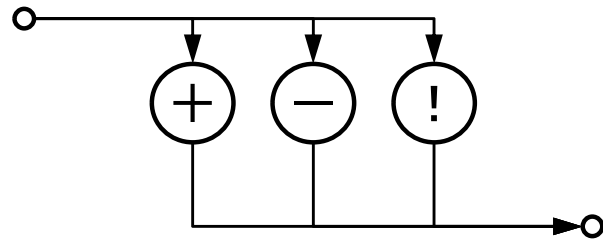
式



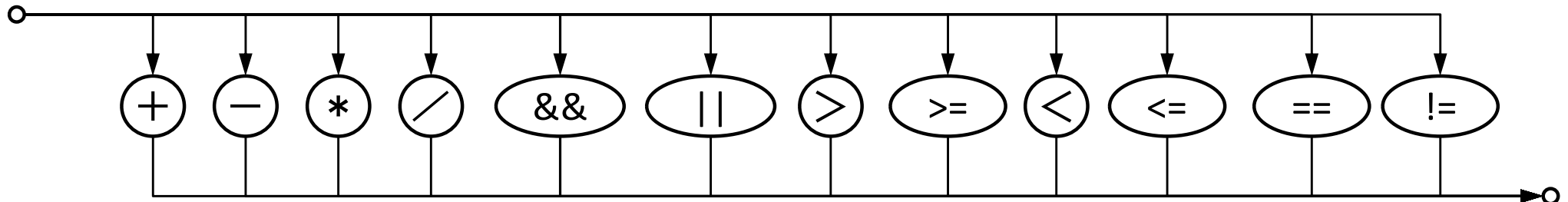
Tiny Cの構文式 ～演算子

- 演算は、整数の算術演算と比較演算、および論理値による論理演算を行うことができる
- 演算子は、単項演算子と二項演算子の二種類がある

単項演算子



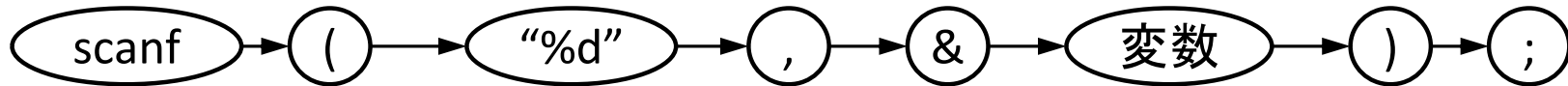
二項演算子



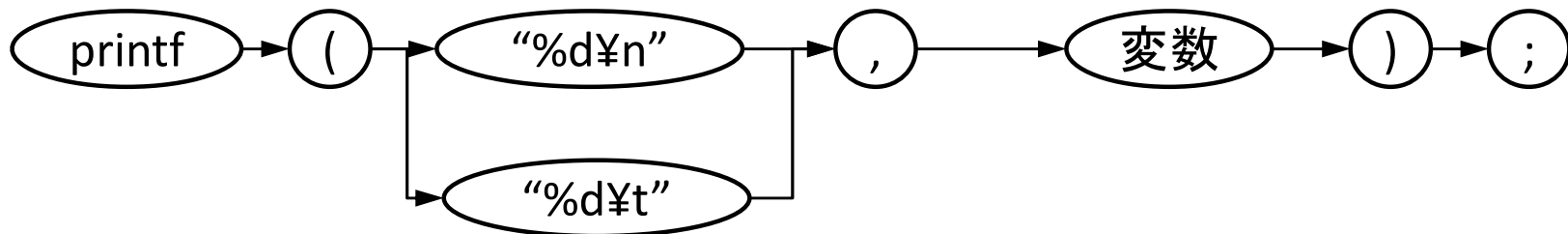
Tiny Cの構文式 ～入出力

- 入出力はscanfとprintfで行う
- Scanfは、一つの非負整数を読み込むことができる
- Printfは、一つの非負整数を出力することができる
 - Printfでは、非負整数を出力した後、改行するか、タブを出力する

入力文



出力文



Tiny Cのサンプルプログラム

【サンプル1】

```
// Display i
int main() {
    int i;

    i = 1;
    printf("%d", i);
}
```

【サンプル2】

```
// input I
int main() {
    int i;

    scanf("%d", &i);
    printf("%d", i);
}
```

【サンプル3】

```
// calculate i
int main() {
    int i;

    scanf("%d", &i);
    i = i * 2;
    printf("%d", i);
}
```

【サンプル4】

```
// complex calculate i
int main() {
    int i;

    scanf("%d", &i);
    i = (i + 1) * 2;
    printf("%d", i);
}
```

【サンプル5】

```
// calculate i
int main() {
    int j;

    scanf("%d", &i);
    i = i * 2;
    printf("%d", i);
}
```

定義されていない変数
No define var

定義されていない演算子
No define operator