

### Переменные

Все переменные имеют целочисленный тип. Считается, что все функции возвращают целочисленный тип.

Процедура инициализации переменных не предусмотрена. ID (имя переменной) - последовательность из букв английского алфавита, цифр и нижнего подчёркивания, не начинающаяся с цифры и не совпадающая с ключевыми словами.

### Конструкции

Цикл

```
while (EXP) { тело цикла }
```

Ввод вывод

```
lout « ID;  
lin » ID;
```

Условные выражения

```
if (EXP) {тело1} [else {тело2}]
```

Вызов функции

```
FUN_ID (имена переменных через запятую);
```

Объявление функции

```
define FUN_ID(аргументы через запятую) {тело1}
```

Присвоение

```
ID := EXP;
```

### Выражения

Выражения бывают двух типов: арифметические (EXP) и логические (LEXP).

EXP = ID | LITERAL | (EXP) | EXP op EXP,

где op из {+, -, \*, /, %}

LEXP = EXP sr EXP | LEXP op LEXP | (LEXP),

где sr из {<, >, <=, >=, ==, !=}, а op из {||, &&}

Выражение, дополненное точкой с запятой, является корректной самостоятельной инструкцией. Например, можно писать так:

```
while (a < 2) { 3 + 3; }
```