

## Задание практикума для 210, 211 гр. (весна 2023)

### «Интерпретатор модельного языка программирования».

Общее описание задания смотрите в методичке «Интерпретатор модельного языка программирования», конкретные требования к выполняемому варианту см. ниже.

#### Требования к реализации:

1. Кроме типов **int** и **string** использовать еще один тип (**real** или **boolean**) на выбор.
2. Набор операторов, который должен быть реализован в модельном языке, следующий:

**<оператор>** →

**if (<выражение>) <оператор> else <оператор>**

**| for ([выражение]; [выражение]; [выражение]) <оператор>**

**| while (<выражение>) <оператор>**

**| break;**

**| goto <идентификатор> ;**

**| read (<идентификатор>);**

**| write (<выражение> { ,<выражение> } );**

**| <составной оператор>**

**| <помеченный оператор>**

**| <оператор-выражение>**

**<помеченный оператор> → <идентификатор>: <оператор>**

Оператор **read** - оператор ввода значения переменной **<идентификатор>;**

оператор **write** – оператор для вывода на экран значений каждого из выражений, указанных в круглых скобках. Форматы ввода и вывода данных определяются реализацией. Числовые константы записываются в десятичной системе счисления.

Семантика остальных операторов как в Си.

3. К общему набору арифметических и логических операций добавить унарный минус.

Граматику для выражений пишете самостоятельно.

## Контекстные условия.

1. Любой идентификатор, используемый в программе, должен быть описан и только один раз.
2. При инициализации переменных типы констант должны совпадать с типами переменных.
3. С помощью оператора **read** можно вводить данные любых типов, определенных в языке, в том числе и логического.
4. С помощью оператора **write** можно выводить значения любых типов, определенных в языке.
5. В операторе цикла **for** типы 1-ого и 3-его выражений произвольные.
6. Если в языке есть логический тип данных (**boolean**), то только логическое выражение может использоваться в условном операторе **if**, в операторах цикла **while** и **for** в качестве условия завершения цикла. Если в языке нет логического типа данных, то используется целочисленное выражение (**0 == false**; любое значение, отличное от **0**, **== true**). Вещественное выражение не может использоваться в качестве условия завершения цикла.
7. Описанием идентификатора-метки считается ее первое использование в программе. Разные операторы не могут быть помечены одинаковыми метками. Одна и та же метка не может помечать один и тот же оператор более одного раза. Метка может находиться как до, так и после оператора **goto**, задающего переход по этой метке.  
  
Для каждого **goto** в программе должен присутствовать соответствующий помеченный оператор. Возможно наличие помеченных операторов, на которые не ведет никакой **goto**.
8. Оператор **break** может встречаться только внутри тела цикла. Появление **break** вне цикла является ошибкой.

Выполнение задания практикума в рамках описанного выше минимального варианта (можно без оператора **break**) при соблюдении требований к синтаксическому и семантическому разбору, указанных в методическом пособии «Интерпретатор

модельного языка», соответствует оценке 3.

Для повышения оценки необходимо расширить программу следующим образом:

1) Добавить возможность задания составного типа

**struct <идентификатор>{<описание>;{<описание>;}};**

доступ к полям структуры должен осуществляться с помощью оператора ‘.’

Контекстные условия:

- Определения структур находятся в начале программы до описаний переменных.

Выделять ли для имен структур отдельное пространство имен ( в этом случае использование перед именем ключевого слова struct будет обязательно) или вводить их в общее пространство имен, решаете сами.

- Допустимые типы полей внутри структуры – только простые (**int, bool, real, string**).

Использование полей составных типов тоже возможно - по желанию.

- Использование значений полей структуры в выражениях такое же, как и у переменных того же типа.

- Возможно присваивание структур целиком, причем только для структур одного типа.

Эквивалентность типов здесь именная, т.е. определяется именем структуры, а не набором ее полей.

Реализация и сдача задания делится на 4 этапа:

1. Лексический анализ;

2. Синтаксический анализ;

3. Семантический анализ + ПОЛИЗ;

4. Интерпретация.