**КОНТЕКСТНО-СВОБОДНАЯ ГРАММАТИКА**

«значение» → «переменная» | «число» | «булевые значения» | «переменная»"."«переменная»| «переменная»“[“«арифметическое выражение»”]“

«идентификатор» → «переменная» | «переменная»"."«переменная» | «переменная» “[“

«арифметическое выражение»”]“

«знак сравнения» → “>” | “<” | “==“ | “!=” | “>=”| “<=”

«арифметический знак» → “+” | “/” | “-“ | “\*” | “%” | “\*\*” | “//”

«логический знак» → “or” | “and“

«булевые значения» → “true“ | “fasle“

«возврат» → "return" ("\n" | «значение» "\n")

«присваивание» → «идентификатор» "=" «арифметическое выражение» "\n" | «идентификатор» "=" «команда чтения» "\n" | «идентификатор» "=" «вызов функции» "\n" | «идентификатор» "=" "[" "]" "\n" | «идентификатор» "=" «определение массива» "\n"

«определение массива» → "["«аргументы»"]"

«определение класса» → “class” «переменная» “(“ “)” “:” “\n” «последовательность»

«определение функций» → "def" «переменная» "(" «аргументы для определения функции» ")" ":" "\n" «последовательность»

«аргументы для определения функции» → «переменная» [“,” | «аргумент для определения функции»] | эпсилон

«аргумент для определения функции» → «переменная» [“,” | «аргумент для определения функции»]

«вызов функции» → «переменная» "(" «аргументы» ")" "\n"

«аргументы» → «идентификатор» [“,” | «аргумент функции»] | | эпсилон

«аргумент» → «идентификатор» [“,” | «аргумент функции»]

«условный оператор if» → “if” «логическая операция» ":" "\n"

«последовательность» [“else” ":" "\n" «последовательность»]

«оператор цикла while» → “while” «логическая операция» ":" "\n"

«последовательность»

«оператор цикла for» → “for” «значение» “in” «значение» ":" "\n"

«последовательность»

«логическая операция» → [“(“] «логическая операция» [«логический знак» «логическая операция»] [“)“]

«логическое выражение» → [“(“] «арифметическое выражение» [«знак сравнения» «логическое выражение»] [“)“]

«арифметическое выражение» →

[“(“]«значение» [«арифметический знак» «арифметическое выражение»] [“)“]

«команда ввода» → “input” “(“ “)”

«команда вывода» → “print” “(“ «арифметическое выражение»” “)” “\n”

«обычное утверждение» → «условный оператор if» | «оператор цикла while» | «оператор цикла for» | «присваивание» | «команда чтения» “\n” | «команда вывода» | «определение класса» | «определение функции» | «вызов функции» | “\n”

«утверждение класса» → «присваивание» | «определение функции» | “\n”

«утверждение функции» → «обычное утверждение» | «возврат»

«блок» → [«таб»] «обычное утверждение» «последовательность» | [«таб»] «утверждение класса» «последовательность» | [«таб»] «утверждение функции» «блок» | эпсилон