NAME: IGE SAMUEL . A

MATRIC NO: 17/SCI01/039

COURSE: CSC 312(Compiler Construction I)

**Question**
Differentiate between lexemes and tokens. Give examples to support your discussion.

**SOLUTION**

1. i. **A lexeme** is a sequence of characters in the source program that matches the pattern for a token and is identified by the lexical analyzer as an instance of that token. WHILE

ii. **A token** is a pair consisting of a token name and an optional attribute value. The token name is an abstract symbol representing a kind of lexical unit, e.g., a particular keyword, or sequence of input characters denoting an identifier. The token names are the input symbols that the parser processes.

1. **Tokens** are symbolic names for the entities that make up the text of the program; e.g. if for the keyword if, and id for any identifier. These make up the output of the lexical analyser. WHILE

 **A lexeme** is a sequence of characters from the input that match a pattern and hence constitute an instance of a token); for example if matches the pattern for if, and foo123bar matches the pattern for id.

1. **Lexeme** - A lexeme is a sequence of characters in the source program that matches the pattern for a token and is identified by the lexical analyzer as an instance of that token. WHILE

 **Token** - Token is a pair consisting of a token name and an optional token value. The token name is a category of a lexical unit.Common token names are:

* Identifiers: names the programmer chooses
* keywords: names already in the programming language
* separators (also known as punctuators): punctuation characters and paired-delimiters
* operators: symbols that operate on arguments and produce results
1. **Token**: The kind for (keywords,identifier,punctuation character, multi-character operators) is ,simply, a Token. WHILE

**Lexeme**: It’s a sequence of characters in SOURCE PROGRAM matched by a pattern for a token. Basically, its an element of Token.

1. **Token:** Token is a sequence of characters that can be treated as a single logical entity. Typical tokens are,
1) Identifiers
2) keywords
3) operators
4) special symbols
5)constants

**WHILE**

**Lexeme:** A lexeme is a sequence of characters in the source program that is matched by the pattern for a token.

**OPRATIONAL REPRESENTATION OF TOKENS AND LEXEMES**

 Lexeme                      Token

 If                                       KEYWORD

 (                                 LEFT PARENTHESIS

 y                                     IDENTIFIER

 < =                                 COMPARISON

 t                                     IDENTIFIER

 )                                RIGHT PARENTHESIS

 y                                    IDENTIFIER

 =                                   ASSGNMENT

 y                                   IDENTIFIER

 \_                                   ARITHMATIC

 3                                     INTEGER