1 Encode

Algorithm 1 Encode algorithm

```
procedure ENCODE
    while input data exists do
        Read_byte
        if there is no matching pattern in the history_buffer the extractly matches the source patern in the holding_buffer, that also satisfiles the condition that offset less 2047 then
            Output_token
        end if
    end while
    Flush
end procedure
```

2 Read_byte

Algorithm 2 Read_byte algorithm

```
procedure READ_BYTE
    Get 8-bit from the input stream
    Insert byte into history_buffer
    Insert byte into holding_buffer
end procedure
```

3 Output_token

Algorithm 3 Output_token algorithm

```
procedure OUTPUT_TOKEN
    if number of bytes in hodling_buffer $\leq 2$ then
        Put single 0 bit to output bit stream
        Put oldest byte in holding_buffer to output stream
        Clear the oldest byte from the holding_buffer
    else if
         thenPut single 1 bit to output bit stream
        if $Offset \leq 127$ then
            Put single 1 bit to output bit stream
            Put 7-bit binary value of $offset$ to output stream
        else if
             thenPut single 0 bit to output bit stream
            Put 11-bit binary value of $offset$ to output stream
        end if
        Output_length
        Clear all bytes from the holding_buffer except the newest byte
    end if
end procedure
```

4 Output_length

Algorithm 4 Output_length algorithm

```
procedure OUTPUT_LENGTH
    $X \leftarrow numberofbytesinholding\_buffer - 1$
    if $X \leq 4$ then
        Put 2-bit binary value of (X-2) to output stream
    else if
        if   thenthen$X \leq 7$
            Put 2-bit pattern with all bits set to a 1bit to output stream
            Put 2-bit binary value of (X-5) to output stream
        else if
             thenPut 4-bit pattern with all bits set to a 1bit to output stream
            $X \leftarrow X - 8$
            while $X \leq 15$ do
                Put 4-bit pattern with all bits set to 1bit to output stream
                $X \leftarrow X - 15$
            end while
        end if
    end if
end procedure
```

5 Offset

Algorithm 5 Offset algorithm

```
procedure OFFSET Return the value of (the distance in bytes within the history_buffer from the first byte of the source pattern to the first byte of the matching pattern).
end procedure
```

6 Flush

Algorithm 6 Flush algorithm

```
procedure Flush
    while number of bytes in holding_buffer $> 0$ do
        Output_token
    end while
    Put 9-bit pattern with b8 and b9 set to 1s and bits b1 through b7 to 0s to output stream
    if desired to clear the history then
        Clear all byte from the history_buffer
    end if
end procedure
```