Name: Rotha Dapravith

ID: e20190915

Group: I5-GIC(B)

**Assignment Discussion Lesson 7**

1) Explain LZ77 compression algorithm?

2) By using LZ77 compression algorithm, find encoder and decoder from 2 different examples? You can choose your own string. Note: First block is more than second block!

**Answers:**

1). LZ77 compression algorithm is used to analyze input data and determine how to reduce the size of that input data by replacing redundant information with metadata. To use it we need:

1) Set the coding position to the beginning of the input stream (first block and second block)

Example: ababbbaaa (first block = 5, second block = 2).

2) Find the longest match in the window for the lookahead buffer. Example: ababbbaaa.

3) If a match is found, output the pointer. Move the coding position (and the window) n bytes forward.

Example: n = 2.

4) If a match is not found, output a null point and the first byte in the lookahead buffer. Move the coding position (and the window) 1 byte forward.

Example: n = 1.

5) If the lookahead buffer is not empty, return to step 2. Unless finish searching.

2). Find encoder and decoder from 2 examples:

Let **abbbccbbbcbaccdab** be the string.

* We choose first 6 blocks: **abbbcc**
* Next 4 blocks be the second: **bbbc**

Step 1:

➢Compare 4 characters from first block with second block.

**a b b b c c** **b b b c** b a c c d a b

* “abbb” ≠ “bbbc” → move 1 character from first block.
* “bbbc” = “bbbc” → match

**a b b b c c** **b b b c** b a c c d a b

6 5 4 3 2 1

We get: Codeword<5, 4, C(b)> (n=4)

Step 2: move n+4 (4+1=5) window at first block

➢Keep taking 6 characters from first block and 4 characters from second block.

**a b b b c c b b b c** **b a c c** d a b

➢Compare 4 characters from first block with second block.

**c c b b b c** **b a** c c d a b

* “ccbb” ≠ “bacc” → move 1 character from first block.
* “cbbb” ≠ “bacc” → move 1 character from first block.
* “bbbc” ≠ “bacc” → move 1 character from first block.
* “So, remove 1 character at the end from second block.
* It rests 3 characters from second block: “dacc”.

➢Compare 3 characters from first block with second block.

**c c b b b c** **b a** c c d a b

* “ccb” ≠ “bac” → move 1 character from first block.
* “cbb” ≠“bac” → move 1 character from first block.
* “bbb” ≠“bac” → move 1 character from first block
* “bbc” ≠“bac” → move 1 character from first block
* So, remove 1 character at the end from second block.
* It rests 2 characters from second block: “ba”.

➢ Compare 2 characters from first block with second block.

**c c b b b c** **b a** c c d a b

* “cc” ≠ “ba” → move 1 character from first block.
* “cb” ≠ “ba” → move 1 character from first block.
* “bb” = “ba” → move 1 character from first block.