Name: VEN THON

ID: e20191250

Group: I5-GIC-C

Assignment Discussion09

1. Explain LZW compression algorithm?

* Encoder algorithm:
* Find all different symbols.
* Initialize the dictionary to contain all strings of length one (build base dictionary).
* Find the longest string W in the dictionary that matches the current input.
* Emit the dictionary index for W to output and remove W from the input.
* Add W followed by the next symbol in the input to the dictionary.
* Go to Step 2 until the last symbol or End of File (EOF)Decoder:
* Decoder algorithm
* Find all different symbols.
* Initialize the dictionary to contain all strings of length one (build base dictionary).
* Find the longest string W in the dictionary that matches the current input.
* Emit the dictionary index for W to output and remove W from the input.
* Add W followed by the next symbol in the input to the dictionary.
* Go to Step 2 until the last symbol or End of File (EOF)

1. By using LZW compression algorithm, find encoder and decoder from 2 different examples? You can choose your own string.

* Example 1: string “aababcaabbac”
* Encoder:

Base Dictionary

|  |  |
| --- | --- |
| Symbol | Codeword |
| a | 1 |
| b | 2 |
| c | 3 |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| s | c | Item | Codeword | Output |
| a | a | aa | 4 | 1 |
| a | b | ab | 5 | 1 |
| b | a | ba | 6 | 2 |
| a | b |  |  |  |
| ab | c | abc | 7 | 5 |
| c | a | ca | 8 | 3 |
| a | a |  |  |  |
| aa | b | aab | 9 | 4 |
| b | b | bb | 10 | 2 |
| b | a |  |  |  |
| ba | c | bac | 11 | 6 |
| c | EOF |  |  | 3 |

**Thus:**  Encoder: { 1 1 2 5 3 4 2 6 3 }

* Decoder: string = { 1 1 2 5 3 4 2 6 3 EOF }

Base Dictionary

|  |  |
| --- | --- |
| Entry | Codeword |
| a | 1 |
| b | 2 |
| c | 3 |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| s | k | Item | Codeword | Output |
| null | 1 |  |  | a |
| a | 1 | aa | 4 | a |
| a | 2 | ab | 5 | b |
| b | 5 | ba | 6 | ab |
| ab | 3 | abc | 7 | c |
| c | 4 | ca | 8 | aa |
| aa | 2 | aab | 9 | b |
| b | 6 | bb | 10 | ba |
| a | 3 | ac | 11 | c |
| c | EOF |  |  |  |

**Thus:**  Decoder: aababcaabbac

* Example 2: string “iluluuiluuu”
* Example 2: string “abcbccabccc”
* Encoder:

Base Dictionary

|  |  |
| --- | --- |
| Entry | Codeword |
| a | 1 |
| b | 2 |
| c | 3 |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| s | c | Item | Codeword | Output |
| a | b | ab | 4 | 1 |
| b | c | bc | 5 | 2 |
| c | b | cb | 6 | 3 |
| b | c |  |  |  |
| bc | c | bcc | 7 | 5 |
| c | a | ca | 8 | 3 |
| a | b |  |  |  |
| ab | c | abc | 9 | 4 |
| c | c | cc | 10 | 3 |
| c | c |  |  |  |
| cc | EOF |  |  | 10 |

**Thus:**  Encoder: { 1 2 3 5 3 4 3 10 }

* Decoder: string = { 1 2 3 5 3 4 3 10 EOF}

Base Dictionary

|  |  |
| --- | --- |
| Entry | Codeword |
| a | 1 |
| b | 2 |
| c | 3 |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| s | k | Item | Codeword | Output |
| null | 1 |  |  | a |
| a | 2 | ab | 4 | b |
| b | 3 | bc | 5 | c |
| c | 5 | cb | 6 | bc |
| bc | 3 | bcc | 7 | c |
| c | 4 | ca | 8 | ab |
| ab | 3 | abc | 9 | c |
| c | 10 | cc | 10 | cc |
| cc | EOF |  |  |  |

**Thus:**  Decoder: abcbccabccc