* Golomb Code ⇒

↳ The Golomb-Rice codes belong to a family of code designed to encode integers with the assumption that the larger an integer, the lower its propendi probability of occurrence.

↳ The simplest code for this situation is the unary code.

↳ The unary code for a positive integer n is simply n 1s followed by a 0.

↳ Thus the code for 4 is 11110, and the code for 7 is 11111110.

↳ The unary code is the same as the Huffman code for the semi-infinite alphabet {1, 2, 3, ...} with probability model, or:

$$\boxed{P[k] = \frac{1}{2^k}}$$

↳ Because the huffman code is optimal, the unary code is also optimal for this probability model.

- The Golomb code is actually a family of codes parameterized by an integer $m>0$.

- In the Golomb code with parameter m, we represent an integer $n>0$ using two numbers q and r, where

$$q = \left\lfloor \frac{n}{m} \right\rfloor$$

and

$$\boxed{r = n - qm}$$

- $\lfloor x \rfloor$ is the integer part of x.

- The quotient q can take on value $0, 1, 2, \ldots$ and is representation by the unary code of q

- The reminder r can take on the values $0, 1, 2, \ldots, m-1$.
- If m is a power of two, can could still use $\lceil \log_2 m \rceil$ bits, where $\lceil x \rceil$ is the smallest integer greater than or equal to x.

Date
Page

* Tunstall Codes :-

↳ Most of the variable-length codes that we look encode letters from the source alphabet using codewords with varying numbers of bits :
Codewords with fewer bits for letters that occur more frequently and codewords with more bits for letters that occur less frequently.

↳ The Tunstall Code is an important exception.

↳ In the tunstall code, all codewords are of equal length.

↳ However, each codeword represent a different numbers of letters.

↳ Example → 2 bit Tunstall code for an alphabet $A = \{A, B\}$ is shown in Table below

Sequence	Codeword
AAA	00
AAB	01
AB	10
B	11

2-bit Tunstall Code

* Advantage ⇒ Advantage of a Tunstall codes is that errors in codewords do not propagate, unlike other variable-length codes, such as Huffman codes, in which an error in one codeword will cause a series of errors to occur.