Outline

PA1 Timing Analysis Report Due Now

Last time:

- Trie
- RLE and Huffman encoding

Today:

- Dictionary-based compression
- LZ77 and LZSS
- LZ78 and LZW

## Dictionary-based Compression

Huffman encoding represents frequently seen symbols with small codes

Often *sequences of symbols* are frequently seen

Example: in English text, “the”, “and”, “ion”, “ing”, “of the”
(and “like”, “so”, “right”) are seen more frequently than other sequences

**Idea:** assign codes to entire sequences

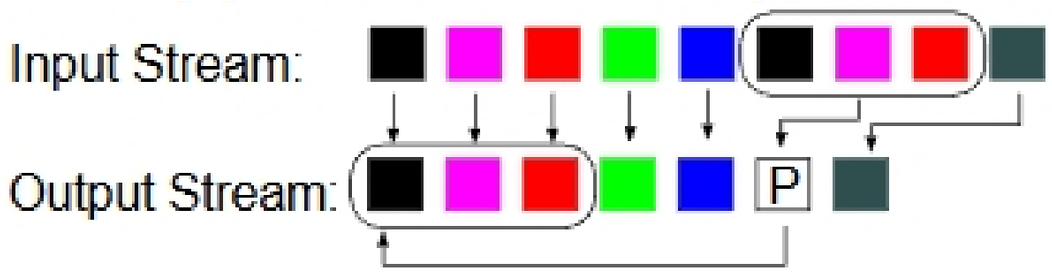
Advantages over Huffman:

- better compression
- **universal code**: encode a piece of text “on-the-fly,”
  in one-pass, “streaming” mode
- no need to store code table (dictionary) in compressed file

## Dictionary-based Taxonomy

All of dictionary-based methods can be divided into two:

- if a sequence has been previously encountered in the input data, output a pointer to the earlier occurrence; the dictionary is implicitly represented by previously processed and output data, example: LZ77




- enter *phrases* into the dictionary; when the same phrase is encountered, output its index into the dictionary, example: LZW