



Why

Can we generalize the idea of flash codes.¹

Definition

Let X be a set and A be an alphabet set. We denote the set of all finite sequences (strings) in A by $\text{str}(A)$. We read $\text{str}(A)$ aloud as “the strings in A .” The length zero string is \emptyset .

A *code* for X in A is a function from X to $\text{str}(A)$. In this context, we refer to the finite set A as an *alphabet* and we call $c(x)$ the *codeword* of x . The *length* of $x \in X$, with respect to a code $c : X \rightarrow \text{str}(A)$, is the length of the sequence $c(x)$ (its codeword). We call a code *nonsingular* if it is injective.

Examples

Define $c : \{\alpha, \beta\} \rightarrow \{0, 1\}$ by $c(\alpha) = (0,)$ and $c(\beta) = (1,)$.²

Code extensions

Let $s, t \in \text{str}(A)$ of length m and n respectively. The *concatenation* of s with t is the length $m + n$ string $u \in \text{str}(A)$ defined by $u_1 = s_1, \dots, u_m = s_m$ and $u_{m+1} = t_1, \dots, u_{m+n} = t_n$. We denote the concatenation of s and t by st . Note, however, that $st \neq ts$, although $s(tr) = (st)r$.

Given a code $c : X \rightarrow \text{str}(A)$, we can produce a code for $\text{str}(X)$ in a natural way. The *extension* of c is the function $C : \text{str}(X) \rightarrow \text{str}(A)$ defined, for $\xi = (\xi_1, \dots, \xi_n) \in \text{str}(X)$, by

$$C(\xi) = c(\xi_1) \cdots c(\xi_n).$$

We call an code *uniquely decodable* if its extension is injective. In other

¹The reliance of this sheet on Flash Codes and Dot-Dash Codes is for this justification, and not for any of the terms presented.

²Future editions will include additional examples.

words, given the code $C(\xi)$ for a sequence $\xi \in \mathbf{str}(X)$, we can recover ξ . We call $C(\xi)$ the *encoding* of ξ . We call ξ the *decoding* of $C(\xi)$.

