Bijective Mapping of Lexicographically Ordered Strings to Natural Numbers

Yahir Josué Ostos Jiménez[†]

Project advisor: M.Sc. Ana Virginia Contreras García[‡]

Abstract. In this article, a bijective mapping is presented between the set of natural numbers $\mathbb{N}$ and the set Σ^* of strings generated by a finite alphabet Σ. This bijection assigns a unique position to each string, thereby inducing a shortlex (length-lexicographic) order through the function. The countability of the set Σ^* is demonstrated, and the importance of this fact is emphasized in applications where an ordered representation of symbol combinations is required. The article also explains how to compute both the absolute position of a string and its relative position within fixed-length subsets, illustrated with concrete examples. These methods provide essential tools for data processing and storage optimization across different areas of application.

1. Introduction. The study of combinations and permutations of symbols within a finite alphabet has been a recurrent topic in areas such as information theory [2], decision theory [14], cryptography [15], and compression algorithms [4]. The need to count and classify character strings in a wide range of applications, from word generation to rewriting [3] to the compact representation of information has motivated the development of methods to systematically and efficiently map [7, 13] an ordered set of strings to natural numbers. In this context, the problem of establishing a bijection between $\mathbb{N}$ and the set formed by all possible strings over a finite alphabet Σ, denoted as Σ^*, arises as an interesting solution for the ordered representation and computational manipulation of combinations of such symbols.

This work aims to construct a bijective function $f : \mathbb{N} \to \Sigma^*$ that preserves the shortlex order in mapping natural numbers to strings a function f is introduced, which assigns to each position $c \in \mathbb{N}$ a unique string in Σ^*, allowing the recovery of both the length and the relative position of each string within a fixed-length set. Additionally, tools are developed for the inverse calculation, obtaining the natural index corresponding to a given string. The construction and proof of this bijection enable an efficient representation of Σ^*, with practical applications in areas such as data compression, combination generation [10, 12], and the design of algorithms that require the shortlex order [5, 8, 9, 17, 19, 20].

This article is organized as follows: In Section 2, some number theory and counting theorems are presented, including their proofs. In Section 3, the function $f : \mathbb{N} \to \Sigma^*$ and its inverse function are constructed. Finally, in Section 4, the use of both functions is presented and the countability of Σ^* is proven.

[†]Facultad de Ingeniería, Universidad Autónoma de Chihuahua (a365335@uach.mx, yahirostos500@gmail.com).
[‡]Facultad de Ingeniería, Universidad Autónoma de Chihuahua (avcontreras@uach.mx).

2. Some Theorems and Definitions. To begin this section, it is very useful to answer the question: How many strings of length m can be written with k distinct symbols? The following theorem answers this question.

Theorem 1 (Number of Strings of Length m Over an Alphabet of Size k). Let Σ be an alphabet containing $k \geq 1$ distinct symbols, and let $m \geq 0$ be an integer. The total number of strings of length m that can be written using the symbols of Σ is k^m.

Proof. Given the alphabet $\Sigma = \{\delta_0, \delta_1, \ldots, \delta_{k-1}\}$, which contains k distinct symbols. We want to determine how many strings of length m can be written using the symbols of Σ.

For $m \geq 1$, each of the m positions has k possible choices. Since the choice of each position is independent of the others, the total number of possible strings of length m is the product of the options available for each position:

$$\text{Total number of strings of length } m = \underbrace{k \times k \times \cdots \times k}_{\text{m times}} = k^m.$$

For $m = 0$, there is exactly one string of length zero, namely the empty string ϵ.

Therefore, the total number of strings of length $m \geq 0$ that can be formed using the symbols of Σ is k^m. ∎

Recall that the set Σ^* is defined as the set of all finite-length strings that can be written using the symbols of the alphabet Σ containing $k \geq 1$ distinct symbols. The following two theorems will help to construct a bijective function $f : \mathbb{N} \to \Sigma^*$.

Theorem 2 (Unique Representation in Base k with m Digits). Let m and k be integers such that $m \geq 1$ and $k \geq 2$, given an integer p such that $0 \leq p \leq k^m - 1$, then exists a unique sequence of m integers $a_{m-1}, a_{m-2}, \ldots, a_1, a_0$; where $0 \leq a_i < k \ \forall i \in \{0, 1, \ldots, m-1\}$, that satisfies:

$$p = a_{m-1}k^{m-1} + a_{m-2}k^{m-2} + \cdots + a_1 k + a_0. \tag{2.1}$$

This sequence $a_{m-1}, a_{m-2}, \ldots, a_1, a_0$ is called the base-k representation of p with m digits.

Proof. Given an integer p such that $0 \leq p \leq k^m - 1$, we seek to prove that p can be uniquely represented as a sequence of m digits in base k, that is, as in equation (2.1), where each a_i is a digit in base k, i.e., $0 \leq a_i < k \ \forall i \in \{0, 1, \ldots, m-1\}$.

- **Existence:** To prove the existence of the representation, a process of successive division is required. We divide the number p by k and apply the Division Algorithm at each step to guarantee that the remainders at each division are in the interval $[0, k-1]$.
  1. **First Division:** Divide p by k and, according to the Division Algorithm, this division gives a quotient p_1 and a remainder a_0, such that:

     $$p = p_1 \cdot k + a_0, \tag{2.2}$$

     where $0 \leq a_0 < k$. The remainder a_0 is the digit at the right end of the base-k representation, and the quotient p_1 is used in the next division.
     Since p is a number between 0 and $(k^m - 1)$, it is clear that p_1 satisfies $0 \leq p_1 < k^{m-1}$.

2. **Second Division:** Now select p_1 and divide it again by k. The Division Algorithm guarantees a new quotient p_2 and a remainder a_1, such that:

$$p_1 = p_2 \cdot k + a_1,$$

where $0 \leq a_1 < k$. The remainder a_1 is the next digit in the base-k representation, and the quotient p_2 is used in the next division.
Since p_1 is between 0 and $(k^{m-1}-1)$, it follows that p_2 satisfies $0 \leq p_2 < k^{m-2}$.
3. **Successive Divisions:** Repeating this process, dividing the quotient p_{i-1} by k at each step. At the i-th step, we obtain the equation:

$$p_{i-1} = p_i \cdot k + a_{i-1}, \tag{2.3}$$

where $0 \leq a_{i-1} < k$, and the quotient p_i is used in the next step. Since $0 \leq p_{i-1} < k^{m-i+1}$, it follows that $0 \leq p_i < k^{m-i}$.
4. **Last Division:** In the last step, the quotient p_{m-1} will be a number less than k, that is, $0 \leq p_{m-1} < k$. Dividing p_{m-1} by k, we get:

$$p_{m-1} = p_m \cdot k + a_{m-1},$$

where $0 \leq a_{m-1} < k$. At this step, p_m must be zero, since p_{m-1} is less than k, which concludes the division process.

From equation (2.3), we have

$$a_{i-1} = p_{i-1} - p_i \cdot k \qquad \text{for } i = 1, \ldots, m.$$

Moreover, by equation (2.2), it follows that $p_0 = p$.
Consider the sum of digits weighted by powers of k:

$$\sum_{j=0}^{m-1} a_j k^j = \sum_{j=0}^{m-1} (p_j - p_{j+1}k)k^j = \sum_{j=0}^{m-1} p_j k^j \; - \; \sum_{j=0}^{m-1} p_{j+1}k^{j+1}.$$

In the second sum change index by setting $t = j+1$. Then $j = 0, \ldots, m-1$ corresponds to $t = 1, \ldots, m$, hence

$$\sum_{j=0}^{m-1} p_{j+1}k^{j+1} = \sum_{t=1}^{m} p_t k^t.$$

Therefore

$$\sum_{j=0}^{m-1} a_j k^j = \sum_{j=0}^{m-1} p_j k^j \; - \; \sum_{t=1}^{m} p_t k^t = p_0 + \sum_{j=1}^{m-1} p_j k^j \; - \; \Big(\sum_{t=1}^{m-1} p_t k^t + p_m k^m \Big).$$

The intermediate sums cancel, leaving

$$\sum_{j=0}^{m-1} a_j k^j = p_0 - p_m k^m.$$

But the algorithm ends with $p_m = 0$ (since $p_{m-1} < k$), so we obtain

$$\sum_{j=0}^{m-1} a_j k^j = p_0 = p.$$

This shows that the digits $a_{m-1}, \ldots, a_0$ satisfy equation (2.1).

- **Uniqueness:** Suppose there exist two distinct sequences

$$a_{m-1}, a_{m-2}, \ldots, a_0 \quad \text{and} \quad a'_{m-1}, a'_{m-2}, \ldots, a'_0$$

with $0 \leq a_i, a'_i < k$ satisfying

$$p = \sum_{i=0}^{m-1} a_i k^i = \sum_{i=0}^{m-1} a'_i k^i.$$

Since the sequences are different, the set $\{\, i \in \{0, \ldots, m-1\} : a_i \neq a'_i \,\}$ is nonempty; let

$$j = \max\{\, i \in \{0, \ldots, m-1\} : a_i \neq a'_i \,\}$$

be the largest index where they differ. By the choice of j, we have $a_i = a'_i$ for every $i > j$. Without loss of generality, assume $a_j > a'_j$; then $1 \leq a_j - a'_j \leq k - 1$. Consider the difference between the two expressions of p:

$$0 = \sum_{i=0}^{m-1} (a_i - a'_i) k^i = (a_j - a'_j) k^j + \sum_{i=0}^{j-1} (a_i - a'_i) k^i.$$

Let

$$R = \sum_{i=0}^{j-1} (a_i - a'_i) k^i.$$

Each term satisfies $|a_i - a'_i| \leq k - 1$, and by the triangle inequality,

$$|R| \leq \sum_{i=0}^{j-1} (k-1) k^i,$$

Then, using the formula for the geometric sum,

$$|R| \leq (k-1) \frac{k^j - 1}{k - 1} = k^j - 1.$$

Using $R \geq -|R|$, we have

$$0 = (a_j - a'_j) k^j + R \geq (a_j - a'_j) k^j - |R|.$$

Then, since $|R| \leq k^j - 1$,

$$0 \geq k^j - (k^j - 1) = 1,$$

which is impossible.
Thus we arrive at a contradiction; consequently, no two distinct sequences can represent p.
We conclude that the sequence of m digits $a_{m-1}, a_{m-2}, \ldots, a_1, a_0$ is unique such that p is represented as in equation (2.1). ∎

The next theorem is the converse of Theorem 2.

Theorem 3. Let m and k be integers such that $m \geq 1$ and $k \geq 2$. Let $a_{m-1}, a_{m-2}, \ldots, a_1, a_0$ be a sequence of m digits, where $0 \leq a_i < k$ for each i. Then, this sequence represents a unique number p in the interval $0 \leq p \leq (k^m - 1)$ given by equation (2.1).

Proof. Given a sequence of m integers, $a_{m-1}, a_{m-2}, \ldots, a_1, a_0$, where each a_i is a digit in the range $0 \leq a_i < k$, a number p is computed using equation (2.1). We want to show that this number p is in the interval $0 \leq p \leq (k^m - 1)$.

Since $a_i \geq 0$ for each i, all terms in the expression for p are non-negative:

$$p = a_{m-1}k^{m-1} + a_{m-2}k^{m-2} + \cdots + a_1 k + a_0 \geq 0.$$

This implies that $p \geq 0$. Moreover, since each digit $a_i \leq (k-1)$, each term in p is bounded, that is,

$$p = a_{m-1}k^{m-1} + a_{m-2}k^{m-2} + \cdots + a_1 k + a_0 \leq (k-1)k^{m-1} + (k-1)k^{m-2} + \cdots + (k-1).$$

Factoring out $k - 1$:

$$p \leq (k-1)(k^{m-1} + k^{m-2} + \cdots + 1).$$

By the geometric sum, then:

$$p \leq (k-1) \cdot \left(\frac{k^m - 1}{k - 1} \right) = k^m - 1.$$

Therefore, p is in the interval $0 \leq p \leq k^m - 1$, which proves the existence of a number p in this interval for each sequence of m integers less than k. ∎

Definition 2.1 (Map σ_m). *Let $\Sigma = \{\delta_0, \delta_1, \ldots, \delta_{k-1}\}$ be an alphabet of size $k \geq 2$, and let A_m be the set containing all strings of length m formed by symbols from Σ. The function*

$$\sigma_m : \{0, 1, \ldots, k^m - 1\} \to A_m$$

is defined as follows: for each number $p \in \{0, 1, \ldots, k^m - 1\}$, express p in base k as a sequence of m digits as in Theorem 2, denoted by

$$(p)_{k_m} = a_{m-1}a_{m-2}\ldots a_1 a_0 \tag{2.4}$$

where each digit $a_i \in \{0, 1, \ldots, k-1\}$. Then define:

$$\sigma_m(p) = \delta_{a_{m-1}}\delta_{a_{m-2}}\ldots\delta_{a_0}. \tag{2.5}$$

Now it is necessary to prove that the map σ_m is well defined.

Theorem 4. The map σ_m from Definition 2.1 is well-defined

Proof. To show that the map σ_m is well defined, we must prove that for every element in its domain, there exists a unique image in A_m.

- **Existence of the Image:**
  Given a number $p \in \{0, 1, \ldots, k^m - 1\}$, it can be represented by the sequence defined by equation (2.4). It is always possible to obtain the string $\delta_{a_{m-1}}\delta_{a_{m-2}} \ldots \delta_{a_0}$ in A_m, by equation (2.5). Therefore, for each p there exists an image in A_m.

- **Uniqueness of the Image:**
  Since the representation of p in base k with m digits is unique by Theorem 2, i.e., $a_{m-1}a_{m-2} \ldots a_1a_0$, the function δ, as defined in equation (2.5), maps each sequence to a unique string $\delta_{a_{m-1}}\delta_{a_{m-2}} \ldots \delta_{a_0}$ in A_m.

Hence, the map σ_m assigns to each p in its domain a unique image in A_m, and therefore it is well-defined. ∎

Theorem 5. The map σ_m from Definition 2.1 is bijective.

Proof. To prove that the map σ_m is bijective, it is necessary to prove that it is injective and surjective.

- **Injectivity of σ_m:**
  Suppose there exist two numbers p and q in the set $\{0, 1, \ldots, k^m - 1\}$ such that $\sigma_m(p) = \sigma_m(q)$; this means that the base k representations of p and q gives the same digit sequence $a_{m-1}, a_{m-2}, \ldots, a_1, a_0$. However, by Theorem 2, the base k representation is unique for each number in the interval $0 \leq p \leq k^m - 1$. Therefore, $p = q$, that shows that σ_m is injective.
- **Surjectivity of σ_m:**
  To show that σ_m is surjective, take any string w of length m in A_m, of the form $\delta_{a_{m-1}}\delta_{a_{m-2}} \ldots \delta_{a_0}$, where each δ_{a_i} is a symbol in Σ. This string corresponds to a digit sequence $a_{m-1}, a_{m-2}, \ldots, a_1, a_0$ with each $a_i \in \{0, 1, \ldots, k-1\}$. By Theorem 3, this sequence represents a unique number p given by equation (2.1) such that $0 \leq p \leq k^m - 1$, and we have $\sigma_m(p) = \delta_{a_{m-1}}\delta_{a_{m-2}} \ldots \delta_{a_0}$, which shows that each string in A_m has a preimage in $\{0, 1, \ldots, k^m - 1\}$. Therefore, σ_m is surjective.

Since σ_m is both injective and surjective, it follows that it is a bijection between the set $\{0, 1, \ldots, k^m - 1\}$ and the set A_m is given by the map σ_m. ∎

It is important to note that that Theorem 5 implies that each number in the domain $\{0, 1, \ldots, k^m - 1\}$ corresponds uniquely to a string in the codomain A_m, and vice versa; which induces a lexicographic ordering on the set A_m.

Since it has already been proven that σ_m is bijective, it follows that its inverse, σ_m^{-1}, satisfies the essential property:

$$\sigma_m(p) = w \iff \sigma_m^{-1}(w) = p, \text{ where } w = \delta_{a_{m-1}}\delta_{a_{m-2}} \ldots \delta_{a_0}.$$

In other words, σ_m^{-1} takes the string $w = \delta_{a_{m-1}}\delta_{a_{m-2}} \cdots \delta_{a_0}$ and reconstructs the corresponding number p whose base-k representation is $a_{m-1}a_{m-2} \cdots a_0$, using equation (2.1).

Definition 2.2 (Function σ_m^{-1}).

Let $\Sigma = \{\delta_0, \delta_1, \ldots, \delta_{k-1}\}$ be an alphabet with $k \geq 1$ symbols, and let A_m denote the set containing all strings of length m formed by the symbols of Σ.

The function $\sigma_m^{-1} : A_m \to \{0, 1, \ldots, k^m - 1\}$ is defined as follows:

$$\sigma_m^{-1}(w) = a_{m-1}k^{m-1} + a_{m-2}k^{m-2} + \cdots + a_1 k + a_0,$$

where $w = \delta_{a_{m-1}}\delta_{a_{m-2}} \ldots \delta_{a_0} \in A_m$ and each $a_i \in \{0, 1, \ldots, k-1\}$.

Theorem 6 (Link between m and S_m). Let k and c be integers such that $k \geq 2$ and $c \geq 1$. We define

$$S_m = \frac{k^{m+1} - 1}{k - 1} = k^0 + k^1 + k^2 + \cdots + k^m, \tag{2.6}$$

then

$$m = \lceil \log_k ((c+1)(k-1) + 1) \rceil - 1 \tag{2.7}$$

if and only if m is the smallest integer such that $c \leq S_m - 1$.

Note that equation (2.6) and equation (2.7) are not defined for $k = 1$ because of the terms involving division by $(k-1)$ and $\log_k$. The case $k = 1$ will be treated separately in Section 3.

Proof. If

$$m = \lceil \log_k ((c+1)(k-1) + 1) \rceil - 1,$$

then m is the smallest integer such that $m \geq \log_k((c+1)(k-1)+1) - 1$, so that

$$k^{m+1} \geq (c+1)(k-1) + 1,$$

which implies

$$c \leq \left(\frac{k^{m+1} - 1}{k - 1} - 1 \right) = S_m - 1,$$

that proves the first part of the theorem.

Suppose that m is the smallest integer satisfying $c \leq S_m - 1$, then

$$c + 1 \leq \frac{k^{m+1} - 1}{k - 1},$$

which, after multiplying by $k - 1$ and adding 1, gives

$$(c+1)(k-1) + 1 \leq k^{m+1}.$$

It implies that m must satisfy

$$m \geq \log_k((c+1)(k-1)+1) - 1.$$

Since m is the smallest value that satisfies this inequality and by properties of the ceiling function, it follows that m satisfies equation (2.7).

It has been shown that

$$m = \lceil \log_k ((c+1)(k-1) + 1) \rceil - 1$$

if and only if m is the smallest positive integer such that $c \leq S_m - 1$, as needed. ∎

In Section 3, Definition 2.1, Theorem 1, Theorem 2, Theorem 3, Theorem 5, and Theorem 6 will be used to construct a function $f : \mathbb{N} \to \Sigma^*$ that is bijective and preserves the shortlex order.

3. Construction of a Bijective Function Preserving the Shortlex Order. The construction of a bijective function $f : \mathbb{N} \to \Sigma^*$ that generates a shortlex order over Σ^* requires three elements: k, the cardinality of Σ; the length m of the string corresponding to the number c (absolute position); and the position p (relative position) within the strings of length m. This is described as follows.

- The first string (at position $c = 0$) is the empty string ϵ, which corresponds to 0 under the bijection.
- Then all strings of length 1 in the order assigned by σ_1, which is lexicographic.
- The next strings are those of length 2 in the order assigned by σ_2, and so on for longer strings.

To assign a string to each $c \in \mathbb{N}$, two components must be determined:

1. The length of the string corresponding to that position c, denoted by m.
2. The relative position p of the number c within the set of strings of length m (that is, within A_m).

For any c, the value of m is the smallest integer such that

$$c \leq S_m - 1 = \left(\frac{k^{m+1} - 1}{k - 1} - 1 \right),$$

where S_m represents the total number of strings of length less than or equal to m. The value of m is obtained from equation (2.7).

The string length m has been determined and it is known that the string w belongs to the set A_m. It is now necessary to determine its relative position within A_m. For this purpose, the value of S_{m-1} must be calculated (using the same notation as in equation (2.6)), which counts all strings of length less than or equal to $m - 1$, including the first string of length m.

The relative position of c within A_m, denoted by p, is given by

$$p = c - S_{m-1}. \tag{3.1}$$

This means that p is the relative position of the string within A_m, and in order to be valid it must be between 0 and $k^m - 1$.

Since $c \geq S_{m-1}$, it follows that

$$p = c - S_{m-1} \geq 0,$$

and since $c \leq S_m - 1$, we have

$$p = c - S_{m-1} \leq S_m - S_{m-1} - 1.$$

Calculating $S_m - S_{m-1}$ yields:

$$S_m - S_{m-1} = k^m,$$

thus:

$$p \leq k^m - 1.$$

Therefore, p is within the interval:

$$0 \leq p \leq k^m - 1,$$

as desired. This ensures that p is a valid integer in the domain of the map σ_m.

Once p has been determined, the map σ_m from Theorem 2.1 can be applied, which by Theorem 5, is a bijective function from the set of integers $\{0, 1, \ldots, k^m - 1\}$ to the set A_m of strings of length m. This allows the string in A_m corresponding to the relative position p to be found:

$$f(c) = \sigma_m(p).$$

The arguments in the previous paragraphs show how the function f is constructed intuitively. A formal definition of f is presented next.

Definition 3.1 (Function f). *Let $\Sigma = \{\delta_0, \delta_1, \ldots, \delta_{k-1}\}$ be an alphabet containing $k \geq 1$ distinct symbols. The function $f : \mathbb{N} \to \Sigma^*$ is defined as follows:*

- ***Case k=1***

$$f(c) = \begin{cases} \varepsilon, & \text{if } c = 0, \\ \delta_0^c, & \text{if } c \geq 1, \end{cases}$$

  where $\delta_0^c = \underbrace{\delta_0, \delta_0, \ldots \delta_0}_{c \text{ times}}$.

- ***Case*** $k \geq 2$

$$f(c) = \begin{cases} \epsilon, & \text{if } c = 0, \\ \sigma_m(p), & \text{if } c \geq 1, \end{cases} \tag{3.2}$$

  where m is given by equation (2.7) and p by equation (3.1).

Theorem 7. The function $f : \mathbb{N} \to \Sigma^*$ defined in Definition 3.1 is well-defined.

Proof. To prove that f is well-defined, we consider separately the cases $k = 1$ and $k \geq 2$.

When $k = 1$, the alphabet is $\Sigma = \{\delta_0\}$ and in both cases, the result is a single string in Σ^*: the empty word when $c = 0$ and a repetition of δ_0 exactly c times when $c \geq 1$. Since each c has a unique corresponding image, the function f is well-defined for $k = 1$.

For $k \geq 2$, we proceed to prove the existence and uniqueness of the image associated with each $c \geq 0$.

- **Existence of the image:** For any $c \geq 0$, the quantity $\log_k((c+1)(k-1)+1)$ can be computed, and the ceiling function $\lceil x \rceil$ ensures that there exists an integer m as given by equation (2.7). This shows that m is well-defined and always exists for every $c \geq 0$. Given m, the value p is defined by $p = c - S_{m-1}$ within the set of strings of length m in Σ^*. The relative position p is computed by means of equation (3.1), which guarantees that p is in the interval $0 \leq p \leq k^m - 1$. The map σ_m provides a bijection between the index set $\{0, 1, \ldots, k^m - 1\}$ and the set A_m of strings of length m, meaning that for each obtained p, the value $\sigma_m(p)$ is a well-defined string in A_m. This ensures that, for each c, there exists an image in Σ^*, since it is always possible to find m, compute p, and obtain a string $w = \sigma_m(p)$ in A_m.

- **Uniqueness of $f(c)$:** To show that $f(c)$ assigns a unique string to each c, it must be verified that both m and p are uniquely determined. First, the ceiling function $\lceil x \rceil$ always assigns a single integer to the value $\log_k((c+1)(k-1)+1)-1$. Thus, for each given c, the value of m is unique. Once m is determined, p is computed uniquely by equation (3.1); thus p is the unique index corresponding to c within the set of strings of length m in Σ^*. As σ_m is a bijection, meaning that each number $p \in \{0, 1, \ldots, k^m - 1\}$ maps uniquely to a string in A_m and vice versa, it follows that the string $f(c) = \sigma_m(p)$ is unique for each c.

The above arguments establish that f is a well-defined function, where each $c \in \mathbb{N}$ has a unique image in Σ^*, completing the proof of existence and uniqueness of f. ∎

Theorem 8. The function f defined in Definition 3.1 is a bijection.

Proof. To prove that f is bijective, it is necessary to show that it is injective and surjective.

When $k = 1$, the function is injective because if $f(c_1) = f(c_2)$, then the corresponding strings $\delta_0^{c_1}$ and $\delta_0^{c_2}$ have the same length, which implies $c_1 = c_2$. It is surjective because every string $w \in \Sigma^*$ is either the empty string $\varepsilon = f(0)$ or of the form $\delta_0^n = f(n)$ for some $n \geq 1$. Therefore, f is bijective when $k = 1$.

We now prove that the function is bijective for $k \geq 2$.

- **Injectivity of f:**
  Suppose that there exist $c_1, c_2 \in \mathbb{N}$ such that $f(c_1) = f(c_2)$. This means that the strings corresponding to c_1 and c_2 in Σ^* are identical. Since each value c yields a unique m and a position p within A_m, if $f(c_1) = f(c_2)$, then necessarily $m_1 = m_2$ and $p_1 = p_2$, due to the bijectivity of σ_m. As m and p are uniquely determined by each c, it must be that $c_1 = c_2$. Therefore, f is injective.
- **Surjectivity of f:**
  Let consider any string $w \in \Sigma^*$. To prove that f is surjective, it is required to find a value $c \in \mathbb{N}$ such that $f(c) = w$. If w is the empty string ϵ, then $f(0) = \epsilon$. Now suppose $w \neq \epsilon$. Let $m \geq 1$ be the length of w, which implies $w \in A_m$, the set of strings of length m in Σ^*. Within the set A_m, the string w has a relative position p, since σ_m is surjective. One may define $c = S_{m-1} + p$, this ensures that the determined value of m from c matches the length of w at position c. By constructing c in such a way it is guaranteed that $f(c) = w$; which means, each string $w \in \Sigma^*$ has a preimage in $\mathbb{N}$, which proves that f is surjective.

Since f is both injective and surjective, it is concluded that f is bijective. ∎

As it has already been proven that f is bijective, its inverse function f^{-1} exists. Let $c \in \mathbb{N}$ and assume $c \geq 1$. Then c can be decomposed according to equation (3.1), where m is the length of the string w associated with c, and p is its relative position in A_m. Then, by the definition of f (Theorem 3.1), we have

$$f(c) = \sigma_m(p) = w.$$

Since σ_m is bijective, its inverse σ_m^{-1} evaluated at w allows the recovery of p (Theorem 2.2):

$$\sigma_m^{-1}(w) = p.$$

Given that $f^{-1}(c) = w$, from equation (3.1) it follows that:

$$f^{-1}(w) = S_{m-1} + p = S_{m-1} + \sigma_m^{-1}(w). \tag{3.3}$$

From equation (3.3), the inverse function of f can be defined.

Definition 3.2 (Inverse function f^{-1}). *Let $\Sigma = \{\delta_0, \delta_1, \ldots, \delta_{k-1}\}$ be an alphabet containing $k \geq 1$ distinct symbols. The function $f^{-1} : \Sigma^* \to \mathbb{N}$ is defined as follows:*

- ***Case*** $k = 1$

$$f^{-1}(w) = \begin{cases} 0, & \text{if } w = \varepsilon, \\ c, & \text{if } w = \delta_0^c \text{ with } c \geq 1. \end{cases}$$

- ***Case*** $k \geq 2$

$$f^{-1}(w) = \begin{cases} 0, & \text{if } w = \varepsilon, \\ S_{m-1} + \sigma_m^{-1}(w), & \text{if } w \neq \varepsilon, \end{cases} \tag{3.4}$$

where m denotes the length of w.

Now we can state this important result

Theorem 9 (Σ^* is countable). Let us consider the alphabet $\Sigma = \{\delta_0, \delta_1, \ldots, \delta_{k-1}\}$, which contains $k \geq 1$ distinct symbols. Then its universe Σ^* is countable.

Proof. Since the function $f : \mathbb{N} \to \Sigma^*$ is bijective, we can conclude that Σ^* is countable. ■

In the following references, similar versions of Theorem 1 [11, 16], Theorem 2 [1], and Theorem 9 [6, 13, 18], along with their corresponding proofs, which are also similar versions to those presented here, can be found.

4. Application examples of the function f. The function f defined in Definition 3.1, along with its inverse, have important applications. One of them is determining which string w appears in position c of Σ^* under a shortlex order.

Let $\Sigma = \{\delta_0, \delta_1, \delta_2\}$ be a finite alphabet. To find the string corresponding to position 52 in Σ^*, it is sufficient to compute $f(52)$. First, determine the length of the string is m, then its relative position p, and finally apply the function σ to find the corresponding string in Σ^*.

- **Step 1: Compute m using the logarithm**
  To compute the length of the string, we use equation (2.7):

$$m = \lceil \log_3((52 + 1) \cdot (3 - 1) + 1) \rceil - 1 = \lceil \log_3(107) \rceil - 1.$$

  Calculating the logarithm:

$$\log_3(107) \approx 4.2533921\ldots,$$

  thus $m = \lceil 4.2533921 \rceil - 1 = 4$. This means that the string at position 52 has length $m = 4$.

- **Step 2: Calculate the relative position p**
  When calculating the total number of strings from length 0 to $m-1=3$, there are 40 strings because:
  $$S_3 = \frac{3^{3+1}-1}{3-1} = \frac{81-1}{2} = 40.$$
  Then, the relative position p within the strings of length 4, given by equation (3.1), is:
  $$p = 52 - 40 = 12.$$
- **Step 3: Convert $p=12$ to base 3**
  Expressing 12 in base 3 gives the sequence of indices of the symbols in Σ (according to the notation in equation (2.4)):
  $$(12)_{3_4} = 0110.$$
  Therefore, the digits 0 and 1 correspond to δ_0 and δ_1, respectively.
- **Step 4: Apply $f(52) = \sigma_m(p)$**
  Hence,
  $$f(52) = \delta_0\delta_1\delta_1\delta_0.$$
  The string at position 52 in Σ^* is $\delta_0\delta_1\delta_1\delta_0$.

Another important application is the inverse process. Suppose we want to compute the position in Σ^* of the string $w = \delta_1\delta_0\delta_2$ within the alphabet $\Sigma = \{\delta_0, \delta_1, \delta_2\}$, which contains $k = 3$ distinct symbols.

Step 1: Determine the length of w The string $w = \delta_1\delta_0\delta_2$ has length $m = 3$.

Step 2: Find the relative position p of the string w within A_m

Since the function σ_m^{-1} gives the relative position p of the string w within the set of strings of length $m = 3$:

1. We must convert the string $w = \delta_1\delta_0\delta_2$ into its index representation. In this case:
   $$w = \delta_1\delta_0\delta_2 \Rightarrow \text{means } (1,0,2) \text{for each index.}$$
2. These indices are interpreted as a number in base 3 using equation (2.1):
   $$p = 1\cdot 3^2 + 0\cdot 3^1 + 2\cdot 3^0 = 1\cdot 9 + 0\cdot 3 + 2\cdot 1 = 9 + 0 + 2 = 11.$$
   Therefore, the relative position of w within A_3 is $p = 11$.

Step 3: Calculate the absolute position c in Σ^* To obtain the absolute position c of the string w in Σ^*, we need to add the total number of strings of length less than $m = 3$, given by S_{m-1}, to the relative position p.

1. We compute S_{m-1} for $m-1=2$:
   $$S_2 = \frac{3^3-1}{3-1} = \frac{27-1}{2} = 13.$$
2. Finally, from equation (3.3), we have:
   $$f^{-1}(w) = S_2 + p = 13 + 11 = 24.$$
   The absolute position in Σ^* of the string $w = \delta_1\delta_0\delta_2$ is $c = 24$.

This example illustrates the process of finding the position of a given string in the shortlex ordered set Σ^* using the inverse function f^{-1}.

5. Conclusion. As mentioned in Section 3, the function $f : \mathbb{N} \to \Sigma^*$ defined in Definition 3.1, together with its inverse function from Definition 3.2, establishes a one-to-one correspondence between the natural numbers and the strings over a finite alphabet Σ, following the shortlex order on Σ^*; within each fixed length this coincides with the usual lex order. This proves the countability of Σ^* (Theorem 9) and has significant practical utility in various fields, from computing to cryptography [2, 4, 14, 15].

The fact that the function is based on well-known theorems (Theorem 2 and Theorem 3) from number theory concerning base-k representations ensures that the concept is easily understood and implemented in a program. This mathematical foundation guarantees that the function is not only theoretically found, but also practically applicable in computational contexts. It enables the mapping and classification of strings, bringing clarity and simplicity to its implementation in software, especially within the design of algorithms that require the shortlex order [7, 10, 12, 13, 19].

Acknowledgments. Thanks to the project advisor, M.Sc. Ana Virginia Contreras García, for her support and valuable suggestions.

References

[1] G. E. Andrews, *Basis representation*, in Number Theory, W. B. Saunders Company, Philadelphia, PA, 1971, pp. 3–8.

[2] R. A. Bourne and S. Parsons, *Connecting lexicographic with maximum entropy entailment*, in European Conference on Symbolic and Quantitative Approaches to Reasoning and Uncertainty, Springer, 1999, pp. 80–91, https://doi.org/10.1007/3-540-48747-6_8.

[3] H. Comon and R. Treinen, *The first-order theory of lexicographic path orderings is undecidable*, Theoretical Computer Science, 176 (1997), pp. 67–87, https://doi.org/10.1016/S0304-3975(96)00049-7.

[4] N. Cotumaccio and N. Prezza, *On indexing and compressing finite automata*, in Proceedings of the 2021 ACM-SIAM Symposium on Discrete Algorithms (SODA), SIAM, 2021, pp. 2585–2599, https://doi.org/10.1137/1.9781611976465.153.

[5] Y. Dong, *Linear algorithm for lexicographic enumeration of cfg parse trees*, Science in China Series F: Information Sciences, 52 (2009), pp. 1177–1202, https://doi.org/10.1007/s11432-009-0132-7.

[6] S. S. Epp, *Regular expressions and finite-state automata*, in Discrete Mathematics with Applications, Cengage Learning, Boston, MA, 5th ed., 2020, pp. 828–858.

[7] P. C. Fishburn and R. L. Graham, *Lexicographic ramsey theory*, Journal of Combinatorial Theory, Series A, 62 (1993), pp. 280–298, https://doi.org/10.1016/0097-3165(93)90049-E.

[8] E. C. Freuder, R. Heffernan, R. J. Wallace, and N. Wilson, *Lexicographically-ordered constraint satisfaction problems*, Constraints, 15 (2010), pp. 1–28, https://doi.org/10.1007/s10601-009-9069-0.

[9] A. M. Frisch, B. Hnich, Z. Kiziltan, I. Miguel, and T. Walsh, *Propagation algorithms for lexicographic ordering constraints*, Artificial Intelligence, 170 (2006), pp. 803–834, https://doi.org/10.1016/j.artint.2006.03.002.

[10] W. Funk and D. J. Lutzer, *Lexicographically ordered trees*, Topology and its Applications, 152 (2005), pp. 275–300, https://doi.org/10.1016/j.topol.2004.10.011.

[11] R. L. Graham, D. E. Knuth, and O. Patashnik, *Number theory*, in Concrete Mathematics: A Foundation for Computer Science, Addison-Wesley Publishing Company, Reading, Massachusetts, 2nd ed., 1994, pp. 102–144.

[12] N. Kemoto, *The lexicographic ordered products and the usual tychonoff products*, Topology and its Applications, 162 (2014), pp. 20–33, https://doi.org/10.1016/j.topol.2013.11.005.

[13] V. Manca, *On the lexicographic representation of numbers*, arXiv preprint arXiv:1505.00458, (2015), https://doi.org/10.48550/arXiv.1505.00458.

[14] M. Mandler, *The lexicographic method in preference theory*, Economic Theory, 71 (2021), pp. 553–577,

https://doi.org/10.1007/s00199-020-01256-2.

[15] S. Rass, A. Wiegele, and S. König, *Security games over lexicographic orders*, in International Conference on Decision and Game Theory for Security, Springer, 2020, pp. 422–441, https://doi.org/10.1007/978-3-030-64793-3_23.

[16] K. H. Rosen, *Counting*, in Discrete Mathematics and Its Applications, McGraw-Hill, New York, 7th ed., 2012, pp. 350–421.

[17] S. Saliba, *Heuristics for the lexicographic max-ordering vehicle routing problem*, Central European Journal of Operations Research, 14 (2006), pp. 313–336, https://doi.org/10.1007/s10100-006-0007-6.

[18] M. Sipser, *Introduction: Automata, computability, and complexity*, in Introduction to the Theory of Computation, Cengage Learning, Boston, MA, 3rd ed., 2013, pp. 1–10.

[19] I. Stojmenovic, *A simple systolic algorithm for generating combinations in lexicographic order*, Computers & Mathematics with Applications, 24 (1992), pp. 61–64, https://doi.org/10.1016/0898-1221(92)90007-5.

[20] A. v. Zanten, *Lexicographic order and linearity*, Designs, Codes and Cryptography, 10 (1997), pp. 85–97, https://doi.org/10.1023/A:1008244404559.