

Exercise:

Prove that the set $S =$

$\{ \pi : \pi \text{ is a permutation of } \{1, 2, 3, \dots, n\}$

for some integer $n \geq 1 \}$

is countable.

CSC 320:

Mathematics of computation

A - Alpha (Al-fah)	N - Nu
B - Beta (Bay-tah)	Ξ - Xi (Zie)
Γ - Gamma	O - Omicron
Δ - Delta	Π - Pi (pie)
E - Epsilon	P - Rho (row)
Z - Zeta	Σ - Sigma
H - Eta (ate-ah)	T - Tau (tahw)
Θ - Theta	Υ - Upsilon
I - Iota (I -oh-ta)	Φ - Phi (fie)
K - Kappa	X - Chi (kie)
Λ - Lamda	Ψ - Psi (sigh)
M - Mu	Ω - Omega



And suddenly there it was, the perfect opening for Tommy's novel, lying at the bottom of his bowl of Alphabet Soup.

Mathematics of Computation

Introduction to the mathematics of formal language theory:

1. Alphabets, strings, languages.
2. Mathematical operations on strings: concatenation, substring, prefix, suffix, reversal.
3. Union, concatenation and Kleene star for languages.

These definitions lead up to our first class of languages- the regular languages.

Review: Representing Data

Alphabet: finite set of symbols

Ex. $\{a, b, c, \dots, z\}$

String: finite sequence of alphabet symbols

Ex. abaab, hello, cccc

Inputs and outputs of computations:
represented by strings.

ε represents an empty string (length 0)

Empty Set and Empty String

Students frequently are confused about the empty set and an empty string.

Empty set = $\Phi = \{ \}$ = set with 0 elements.

Empty string = $\varepsilon = ''$ = string with 0 symbols.

Example:

The set $\{\varepsilon\}$ is a set containing one string.

Language: set of strings

First names of students taking CSC 320:

{Adrian, Barry, Bernadette, Chiu, Christopher,
Cory, **Daniel**, **Derek**, Devin, Emily, Eric, Erik, Evan,
Fan, Gareth, Gregory, James, Jeremy, Jesse, Jessie,
Justin, Kahlil, Kelvin, Laura, Lin, Ling-Yu, Lucas,
Malcolm, Matthew, Maxwell, Michael, Nathan,
Nazma, Nicholas, Nikolas, Pauline, Riley, **Ryan**,
Sahibdeep, Simon, Thomas, Troy}

Strings over {a,b} with even length

{ ϵ , aa, ab, ba, bb, aaaa, aaab, aaba, ...}

Syntactically correct JAVA programs

Notational Conventions

x, y, z real numbers

$\Sigma, \Sigma', \Sigma_1, \Sigma_2$
alphabets

k, n integers

$a, b, c, 0, 1$ symbols

A, B matrices

u, v, w, x, y, z strings

p, q primes

L, L_1, L_2 languages

Concatenation of strings x and y denoted $x \cdot y$ or simply xy means write down x followed by y .

Theorem: For any string w , $\epsilon \cdot w = w = w \cdot \epsilon$.

String v is a **prefix** of w if $w = v y$ for some string y .

String v is a **suffix** of w if $w = x v$ for some string x .

String v is a **substring** of w if there are strings x and y such that $w = x v y$.

The **reversal** of string w , denoted w^R , is w "spelled backwards".

Operations on Languages:

1. **Complement** of L defined over $\Sigma = \overline{\Sigma}$
 $= \{ w \in \Sigma^* : w \text{ is not in } L \}$
2. **Concatenation** of Languages $L_1 \cdot L_2 = L_1 L_2 =$
 $\{w = x \cdot y \text{ for some } x \in L_1 \text{ and } y \in L_2\}$
3. **Kleene star** of L , $L^* = \{ w = w_1 w_2 w_3 \dots w_k \text{ for}$
some $k \geq 0$ and $w_1, w_2, w_3, \dots, w_k$ are all in $L \}$
4. $L^+ = L \cdot L^*$

(Concatenate together one or more strings from L .)

Σ^* = set of all strings over alphabet Σ

Language over Σ - any subset of Σ^*

Examples: $\Sigma = \{0, 1\}$

$L_1 = \{ w \in \Sigma^* : w \text{ has an even number of 0's} \}$

$L_2 = \{ w \in \Sigma^* : w \text{ is the binary representation of a prime number with no leading zeroes} \}$

$L_3 = \Sigma^*$

$L_4 = \{ \} = \Phi$

$L_5 = \{ \varepsilon \}$

$L_2 = \{w \in \{0,1\}^* : w \text{ is the binary representation of a prime with no leading zeroes}\}$

The complement is:

$\{w \in \{0,1\}^* : w \text{ is the binary representation of a number which is not prime which has no leading 0's or } w \text{ starts with 0}\}$

Note: 1 is not prime or composite. The string 1 is in the complement since it is not in L .

Matrix multiplication:

$$\begin{bmatrix} 1 & 2 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 1 \\ 1 & 1 \end{bmatrix} = \begin{bmatrix} 3 & 3 \\ 1 & 1 \end{bmatrix}$$

$$\begin{bmatrix} 1 & 1 \\ 1 & 1 \end{bmatrix} \begin{bmatrix} 1 & 2 \\ 0 & 1 \end{bmatrix} = \begin{bmatrix} 1 & 3 \\ 1 & 3 \end{bmatrix}$$

Concatenation:

$$ab \cdot bb = abbb$$

$$bb \cdot ab = bbab$$

Regular Languages over Alphabet Σ :

[Basis] 1. Φ and $\{\sigma\}$ for each $\sigma \in \Sigma$ are regular languages.

[Inductive step] If L_1 and L_2 are regular languages, then so are:

2. $L_1 \cdot L_2$,

3. $L_1 \cup L_2$, and

4. L_1^* .

Regular expressions over Σ :

[Basis] 1. Φ and σ for each $\sigma \in \Sigma$ are regular expressions.

[Inductive step] If α and β are regular expressions, then so are:

2. $(\alpha\beta)$

3. $(\alpha \cup \beta)$ and

4. α^*

Note: Regular expressions are strings over

$\Sigma \cup \{ (,), \Phi, \cup, * \}$

for some alphabet Σ .

Precedence of Operators

Exponents

highest

Kleene star

Multiplication



Concatenation

Addition

lowest

Union

Prove the following languages over $\Sigma=\{0,1\}$ are regular by giving regular expressions for them:

1. $\{w: w \text{ has odd length}\}$
2. $\{w: w \text{ contains } 0011\}$
3. $\{w: w \text{ does not contain } 01\}$
4. $\{w: w \text{ starts and ends with the same symbol } (|w| \geq 1)\}$

TUTORIAL: $L = \{ w \text{ an element of } \{a,b\}^* : w \text{ has both baa and aaba as a substring} \}$.

$(a|b)^*baa(a|b)^*aaba(a|b)^*|(a|b)^*aaba(a|b)^*baa(a|b)^*$

$L = \{ w \text{ an element of } \{a,b\}^* : w \text{ has both baa and aaba as a substring} \}$.

Give a regular expression for L.

Your Answer

Your answer should generate the following strings but does not
aabaa, baaba, aaabaa, aabaaa, abaab, abaaba, baaaba, baabaa,
baabab, bbaaba, aaaabaa, aaabaaa, aaabaab, aabaaaa, aabaaab,
aabaaba, aabaabb, abaaaba, abaabaa, abaabab

Lesson	Syntax	Hint	Answer
Previous Question	Next Question	Submit	

MISSING:

aabaa,

baaba,

...

See home
page for link
to regular
expression
tutorial