Tutorial 01

University of Victoria

CSC 320 - Spring 2023

Foundations of Computer Science

Teaching Team

Learning Outcomes:

- Remember essential mathematical concepts.
- Become familiar with countability.
- Become familiar with set theory.
- Become familiar with languages.

Interesting Article:

"Undergraduates' Example Use in Proof Construction: Purposes and Effectiveness" [1]

January 17th, 2023

Outline

A review on essential mathematical concepts and an overview on languages.

1. **Countability**: A brief overview.
2. **Set Theory**: A review of set theory and describing sets.
3. **Proof Types**: A review on proof by contradiction, contrapositive, induction, and construction.
4. **Languages**: A review on alphabets, symbols, strings, etc..

Countability

We know that there are three types. Finite, Countably Infinite, and Uncountably Infinite.

Finite means that you can count the elements up to some number n.

Countably Infinite maps to N, the set of natural numbers.

Uncountably Infinite means there exists no way of counting that maps to N, the set of natural numbers.

subsection 8.4 - Countable Sets and Sequences in Dr. Gary's Notes [2] is a good resource if you need further examples.

Set Theory

A Few Basic Definitions

- set: $S = \{a, b, c, d\}$
- membership: $a \in S$, $f \notin S$
- empty set: $\emptyset$
- singleton set: set with exactly 1 element
- unordered pair set: set with 2 elements

subsection 0.2 - Mathematical Notions And Terminology in Introduction to the Theory of Computation is a good resource if you need further examples.

A Few Basic Definitions

- union: $A \cup B$
- intersection $A \cap B$
- complement: $\bar{A}$
- set difference: $A - B$
- cartesian / cross product: $A \times B$

subsection 6.1 - Cartesian Products in Dr. Gary's Notes [2] is a good resource if you need further examples.

Describing Sets

$$\{x \mid x = 2m \text{ for each m in } \mathbb{N},\ m > 5\}$$

Writing Sets

- The set of all integers greater than 5.
- The set of all strings 0...01...1 where all 0's come before 1's and there are twice as many 0's as 1's.
- The set of all odd number ≥ 1.

Proof Types

I highly recommend reading subsection 0.4 - Types of Proof in Introduction to the Theory of Computation. It goes over contradiction, induction, and construction.

Contradiction

Prove that $\sqrt{2}$ is an irrational number by contradiction.

Remember: You'll want to state your goal (i.e., prove that $\sqrt{2}$ is an irrational number). You'll then want to assume the opposite (i.e., assume that $\sqrt{2}$ is a rational number). And finally, you'll want to draw a conclusion that results in a contradiction (i.e., a statement that contradicts $\sqrt{2}$ is a rational number). Lastly, you'll need to write a formal conclusion.

Contrapositive

Given $p \longrightarrow q$, prove $\neg q \longrightarrow \neg p$.

Prove that for any integer n, if n^2 is even then n is even.

1. Suppose n is odd.
2. Prove that n^2 is odd.

subsection 1.7 - Converse and Contrapositive of an Implication in Dr. Gary's Notes is a good resource if you need further examples.

Induction

Prove that $1 + 2 + ... + n = n(n + 1)/2$.

1. Prove base case(s).
2. State inductive hypothesis.
3. Perform inductive step.

4. Write conclusion.

Construction

Theorems often state that an object exists. Proof by Construction is a way to prove the theorem by constructing the object (i.e., proof by look here's one).

Example: Prove that there is a program that can be used to calculate $A + B$. Solution: Write a program that can calculate $A + B$.

Languages

Very Informally

- alphabets and languages will be key components of this course.
- alphabets are a set of symbols, just as you know from daily life.
  - for example, the alphabet for English is a, b, c, ..., z
  - while the alphabet for German is a, b, c, ..., β, z, ü, ö, ä
- similarly, language are a set of words (strings) made up from a given alphabet.
  - for example, the English language contains words like "hello", "goodbye"
  - the French language contains words like "bonjour", "chat"

Note: Even when given the exact same alphabet, you can create different languages.

subsection 0.2 - Mathematical Notions And Terminology in Introduction to the Theory of Computation is a good resource if you need further examples.

Formally

- an alphabet is a finite set of symbols denoted Σ
- a string is any combination of symbols in Σ
- empty string = ϵ
- set of all strings of an alphabet is Σ^*
- length of strings, $|\epsilon| = 0$, if $w = ab$, then $|w| = 2$
- position, $w = aba$, then $w_2 = b$, and $w_1 = w_3 = a$
- concatenation, $x = hi$ and $y = bye$, then $xy = hibye$

Languages

- a language is a set of strings
- Kleene star, L^*
  - concatenate all substrings of L with L (infinite)

Example $L_1 = \{00, 11\}$

$$L_1^* = \{\epsilon, 00, 11, 0011, 1100, ...\}$$

where $*$ means 0 or more occurrences.

$$L_1^+ = \{00, 11, 0011, 1100, ...\}$$

where $+$ means at least 1 occurrence.

Tutorial 02

University of Victoria

CSC 320 - Spring 2023

Foundations of Computer Science

Teaching Team

Learning Outcomes:

- Become familiar with DFAs and NFAs.
- Become familiar with the concept of Closure.
- Become familiar with the concept of Kleene Star.
- An introductory level of understanding of Reduction.

Interesting Article:

"On Theory of Regular Languages with the Kleene Star Operation" [3]

January 24th, 2023

Question 2.01

Give the formal specification of a DFA for the following language:

$$L = \{0\}^* \text{ over } \Sigma = \{0\}$$

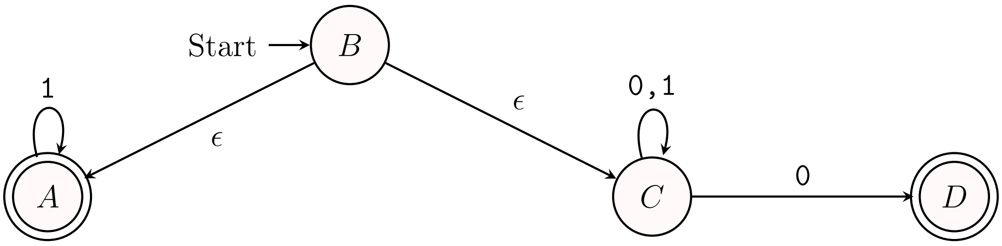
Question 2.02

Give the formal specification of a DFA for the following language:

$$L = \{w \in \{a, b\}^* \mid w \text{ is any string not in } (ab^+)^*\}$$

Question 2.03

Consider the following state diagram:



where the start state is B, and state A and state D are accept states. And can be describe by the following transition table:

δ	0	1	ϵ
A	$\emptyset$	A	B
B	$\emptyset$	$\emptyset$	$\{A, C\}$
C	$\{C, D\}$	C	$\emptyset$
D	$\emptyset$	$\emptyset$	$\emptyset$

Table 1: State Diagram - Transition Table

(a) Is the string 0011 accepted by this state machine? How about 1100?

(b) What is the language of this machine?

Question 2.04

Design an NFA state diagram for the following language:

$$\{w \in \{0,1\}^* \mid w \text{ contains 00 or 11 as a substring}\}$$

DFA Union Closure

Regular languages are closed under union.

What does "closed" mean?

A set S is closed under operation O if $O(S) \in S$.

Let $S = \{a, b, c\}$. Define O as such: $O(a) = b$, $O(b) = c$, and $O(c) = a$.

Notice that applying O yields elements that are all in set S. So S is closed under O. If O were defined the same but $O(c) = z$, then S is no longer closed under O.

Kleene Star Proof

Prove that regular languages are closed under Kleene star.

Reduction Discussion

Reduction...

Problem A: Will Ammar Brush His Hair?

Problem B: Is Angela Happy?

Reduction:

$$A \longrightarrow B \text{ "A reduces to B"}$$

The outcome of A relies on the outcome of B.

Tutorial 03

University of Victoria

CSC 320 - Spring 2023

Foundations of Computer Science

Teaching Team

Learning Outcomes:

- Design a regular expression for a language.
- Convert a regular expression to an NFA.
- Convert a DFA to a regular language.

Interesting Article:

"Compressing Regular Expressions' DFA Table by Matrix Decomposition" [4]

January 31st, 2023

Question 3.01

Design a regular expression for the following languages over the alphabet $\Sigma = \{0, 1\}$:

(a) $L_1 = \{w \mid$ **every odd position of** w **is a** $1\}$

(b) $L_2 = \{w \mid w$ **is a string of length at most** $5\}$

(c) $L_3 = \{w \mid w$ **contains an even number of** 0**'s or exactly two** 1**'s**$\}$

Question 3.02

Convert the following regular expression to an NFA:

$$R_1 = (a \cup b^*)a$$

DFA to Regular Expression

If a language is regular, then there exists some regular expression that describes it.

Transform the following DFA into a Regular Expression:

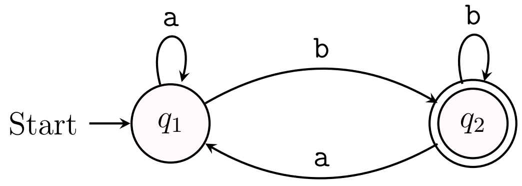


Figure 1: DFA

where the DFA can be describe by the following:

$$DFA = (\{q_1, q_2\}, \{a, b\}, \delta, q_1, \{q_2\})$$

and δ is defined as:

δ	a	b
q_1	q_1	q_2
q_2	q_1	q_2

Table 2: DFA - Transition Table

Tutorial 04

University of Victoria

CSC 320 - Spring 2023

Foundations of Computer Science

Teaching Team

Learning Outcomes:

- Understand the Pumping Lemma.
- Prove that a language is not regular using the Pumping Lemma.

Interesting Article:

"Pumping Lemma for Quantum Automata" [5]

February 7th, 2023

Question 4.01

Prove that the following language is not regular using the pumping lemma.

$$L_1 = \{0^n 1^n 2^n \mid n \geq 0\}$$

Question 4.02

Prove that the following language is not regular using the pumping lemma.

$$L_2 = \{w^r w \mid w \in \{0,1\}^*\}$$

Reflection on Question 4.02

Why is the string $s = 0^P 0^P$ not a good choice to devise a contradiction to prove L_2 is not regular?

Tutorial 05

University of Victoria

CSC 320 - Spring 2023

Foundations of Computer Science

Teaching Team

Learning Outcomes:

- Become familiar with Context Free Grammars.
- Convert a Context Free Grammar into Chomsky Normal Form.
- Use a Pushdown Automata to describe a language.

Interesting Article:

"A Formalisation of the Cocke-Younger-Kasami Algorithm" [6]

February 14th, 2023

Question 5.01

Consider the following language over $\Sigma = \{0, 1\}$, find a set of rules that defines a Context Free Grammar (CFG) that recognizes the language:

$$L_1 = \{0^n 1^n \mid n \geq 0\} \cup \{1^n 0^n \mid n \geq 1\}$$

Question 5.02

Consider the following language over $\Sigma = \{0, 1\}$, find a set of rules that defines a CFG that recognizes the language:

$$L_2 = \{w \mid w \text{ starts and ends with the same symbol}\}$$

Question 5.03

Consider the following language over $\Sigma = \{0, 1\}$, find a set of rules that defines a CFG that recognizes the language:

$$L_3 = \emptyset$$

Question 5.04

Consider the following language over $\Sigma = \{0, 1\}$, find a set of rules that defines a CFG that recognizes the language:

$$L_4 = \{w \mid w \text{ contains at least three 1s}\}$$

Question 5.05

Consider the following language over $\Sigma = \{0, 1\}$, find a set of rules that defines a Context Free Grammar (CFG) that recognizes the language:

$$L_5 = \{0^n 1^m \mid 2n \leq m \leq 3n\}$$

Question 5.06

Consider the following language over $\Sigma = \{0, 1\}$, create a parse tree and show sequence derivations for the following string: 000111

$$L_6 = \{0^n 1^n \mid n \geq 0\}$$

Question 5.07

Prove or Disprove: Every subset of a Context Free Language (CFL) is a regular language.

Question 5.08

Convert the following CFG into Chomsky Normal Form:

$$
\begin{aligned}
S &\longrightarrow ASB \\
A &\longrightarrow aAS \mid a \mid \epsilon \\
B &\longrightarrow SbS \mid A \mid bb
\end{aligned}
$$

Question 5.09

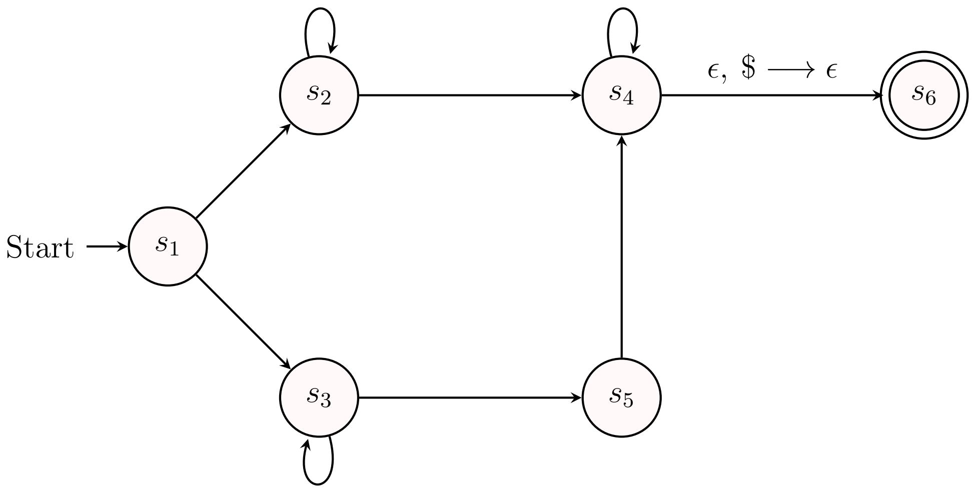
Convert the following CFG into Chomsky Normal Form:

$$
\begin{aligned}
S &\longrightarrow aXbX \\
X &\longrightarrow aY \mid bY \mid \epsilon \\
B &\longrightarrow X \mid c
\end{aligned}
$$

Question 10

Complete the state diagram by adding missing transitions so that it describes a PDA that recognizes the following language:

$$L = \{a^m b^n \mid m, n \geq 0 \text{ and (either } m = n \text{ or } m = n + 2)\}$$



where the PDA can be describe by the following:

State	Input Symbol	Stack Symbol	Next State	Stack Operation
s_1			s_2	
s_1			s_3	
s_2			s_2	
s_2			s_4	
s_3			s_3	
s_3			s_5	
s_4			s_4	
s_4	ϵ	\$	s_6	ϵ
s_5			s_4	

Table 3: PDA - Transition Table

Question 11

Derive or generate the string "aabaa" for the following grammar:

$$
\begin{aligned}
S &\longrightarrow aAS \mid aSS \mid \epsilon \\
A &\longrightarrow SbA \mid ba
\end{aligned}
$$

Question 12

Convert the following CFG into Chomsky Normal Form:

$$
\begin{aligned}
S &\longrightarrow AAA \mid \epsilon \\
A &\longrightarrow aa \mid Aa \mid \epsilon
\end{aligned}
$$

Tutorial 06

University of Victoria

CSC 320 - Spring 2023

Foundations of Computer Science

Teaching Team

Learning Outcomes:

- Become familiar with Context Free Languages.
- Use the Context Free Language Pumping Lemma.
- Gain understanding of High Level description of a Turing Machine.

Interesting Article:

"Regular Patterns, Regular Languages and Context-Free Languages" [7]

February 28th, 2023

Question 6.01

Show that the following language is not CFL using the CFL pumping lemma:

$$A = \{a^i b^j c^k \mid 0 \leq i \leq j \leq k\}$$

Question 6.02

Show that the following language is not CFL using the CFL pumping lemma:

$$B = \{0^n \# 0^{2n} \# 0^{3n} | n \geq 0\}$$

Question 6.03

Give a high level description of a Turing Machine which decides:

$$C = \{0^{2^n} \mid n \geq 0\}$$

Hint: Review page 143 in the Textbook.

Tutorial 07

University of Victoria

CSC 320 - Spring 2023

Foundations of Computer Science

Teaching Team

Learning Outcomes:

- Construct High Level Turing Machines.
- Distinguish between different Turing Machine variants.
- Become familiar with Deciders vs Looping.

Interesting Article:

"Note on A Universal Quantum Turing Machine" [8]
"Quantum Chaos in Quantum Turing Machines" [9]

March 7th, 2023

Question 7.01

Construct a PDA from the following CFG:

$$S \longrightarrow S1 \mid 1S0S \mid \epsilon$$

Question 7.02

Give a high level description of a Turing Machine which decides:

$$A = \{0^{2^n} \mid n \geq 0\}$$

Question 7.03

Give a high level description of a Turing Machine which decides:

$$B = \{a^i b^j c^k \mid i \times j = k \text{ and } i, j, k \geq 1\}$$

Question 7.04

Give a high level description of a 2-tape TM which recognizes the language (binary palindromes):

$$L = \{w \in \{0,1\}^* \mid w = w^r\}$$

Question 7.05

Give a high level description of a nondeterministic TM which recognizes the language:

$$L = \{1^n \mid n \text{ is a composite number}\}$$

Question 7.06

Prove that the following language is decidable by constructing (high level) Turing Machines which decide the language.

$$A_{DFA} = \{\langle B, w\rangle \mid B \text{ is a DFA that accepts input string } w\}$$

Question 7.07

Prove that the following language is decidable by constructing (high level) Turing Machines which decide the language.

$$E_{DFA} = \{\langle A \rangle \mid A \text{ is a DFA and } L(A) = \emptyset\}$$

Question 7.08

Prove that the following language is decidable by constructing (high level) Turing Machines which decide the language.

$$EQ_{DFA} = \{\langle A, B\rangle \mid A \text{ and } B \text{ are DFAs and } L(A) = L(B)\}$$

Tutorial 08

University of Victoria

CSC 320 - Spring 2023

Foundations of Computer Science

Teaching Team

Learning Outcomes:

- Construct high level Turing Machines.
- Prove that a language is decidable by construction.

Interesting Article:

"Non-Erasing Turing Machines: A New Frontier Between A Decidable Halting Problem and Universality" [10]
"Strongly Universal Quantum Turing Machines and Invariance of Kolmogorov Complexity" [11]

March 14th, 2023

Question 8.01

Prove that the following language is decidable by constructing (high level) Turing Machines which decide the language.

$$A_{DFA} = \{\langle B, w\rangle \mid B \text{ is a DFA that accepts input string } w\}$$

Question 8.02

Prove that the following language is decidable by constructing (high level) Turing Machines which decide the language.

$$E_{DFA} = \{\langle A \rangle \mid A \text{ is a DFA and } L(A) = \emptyset\}$$

Question 8.03

Prove that the following language is decidable by constructing (high level) Turing Machines which decide the language.

$$EQ_{DFA} = \{\langle A, B\rangle \mid A \text{ and } B \text{ are DFAs and } L(A) = L(B)\}$$

Tutorial 09

University of Victoria

CSC 320 - Spring 2023

Foundations of Computer Science

Teaching Team

Learning Outcomes:

- Use reduction to prove a language is undecidable.
- Become familiar with reduction.

Interesting Article:

"Decidable and Undecidable Problems about Quantum Automata" [12]

March 21st, 2023

Question 9.01

Prove that the following language is undecidable by reduction from A_{TM}.

$$A_{TM} = \{\langle M, w\rangle \mid M \text{ is a TM and } M \text{ accepts } w\}$$

$$\text{Regular}_{TM} = \{\langle M\rangle \mid M \text{ is a TM and } L(M) \text{ is a regular language}\}$$

Question 9.02

Prove that the following language is undecidable by reduction from A_{TM}.

$$A_{TM} = \{\langle M, w\rangle \mid M \text{ is a TM and } M \text{ accepts } w\}$$

$$S_{TM} = \{\langle M\rangle \mid M \text{ is a TM that accepts } w^r \text{ whenever it accepts } w\}$$

Question 9.03

Prove that the following language is undecidable by reduction from A_{TM}.

$$A_{TM} = \{\langle M, w\rangle \mid M \text{ is a TM and } M \text{ accepts } w\}$$

$$S_{TM} = \{\langle A\rangle \mid A \text{ is a DFA and } L(A) = \emptyset\}$$

Question 9.04

Prove that the following language is undecidable by reduction from A_{TM}.

$$A_{TM} = \{\langle M, w\rangle \mid M \text{ is a TM and } M \text{ accepts } w\}$$

$$E_{TM} = \{\langle M\rangle \mid M \text{ is a TM and } L(M) = \emptyset\}$$

Question 9.05

Prove that the following language is undecidable by reduction from ALL_{CFG}.

$$ALL_{CFG} = \{\langle G \rangle \mid G \text{ is a CFG and } L(G) = \Sigma^*\}$$

$$EQ_{CFG} = \{\langle G, H \rangle \mid G \text{ and } H \text{ are CFGs and } L(G) = L(H)\}$$

Tutorial 10

University of Victoria

CSC 320 - Spring 2023

Foundations of Computer Science

Teaching Team

Learning Outcomes:

- Become familiar with NP Membership.
- Review and understand verifiers and certificates.
- Write proofs using reduction.

Interesting Article:

"An Improved Exact Algorithm for Minimum Dominating Set in Chordal Graphs" [13]

March 28th, 2023

Question 10.01

NP-Completeness - Polynomial Time Reductions

Given an input $\langle G, k \rangle$ for Clique, where $G = (V, E)$ is an undirected graph and k is a positive integer. Use reduction to prove the following:

$$\text{Clique} \leq_{\mathrm{p}} \text{Independent Set (IS)}$$

Question 10.02

NP-Completeness - Polynomial Time Reductions

Given an input $\langle G, k \rangle$ for IS, where $G = (V, E)$ is an undirected graph and k is a positive integer. Use reduction to prove the following:

$$\text{Independent Set (IS)} \leq_{\text{p}} \text{Vertex Cover (VC)}$$

Question 10.03

NP-Completeness - Polynomial Time Reductions

Let $\langle G, k \rangle$, $G = (V, E)$ and $k \in \mathbb{N}$, be an instance for VC. Use reduction to prove the following:

$$\text{Vertex Cover (VC)} \leq_{\text{p}} \text{Dominating Set (DS)}$$

for graphs without singletons.

Question 10.04

NP-Completeness - Polynomial Time Reductions

Given an input $\langle G, k \rangle$ for VC, where $G = (V, E)$ is an undirected graph and k is a positive integer. Use reduction to prove the following:

$$\text{Vertex Cover (VC)} \leq_{\text{p}} \text{Independent Set (IS)}$$

Question 10.05

NP-Completeness - Polynomial Time Reductions

Given an input $\langle G, k \rangle$ for IS, where $G = (V, E)$ is an undirected graph and k is a positive integer. Use reduction to prove the following:

$$\text{Independent Set (IS)} \leq_{\text{p}} \text{Clique}$$

Question 10.06

Membership in NP - Verifiers

We remember that Nondeterministic Polynomial Time (NP) is the set of languages for which there exists a polynomial time verifier. Thus, we can prove something is in NP by writing a verifier!

Verifiers

Verifiers take a problem and a potential solution C and check if C is actually a solution or not.

Consider the problem GraphColouring...

Question 10.07

Membership in NP - Verifiers

Prove CLIQUE is in NP.

$$\text{CLIQUE} = \{\langle G, k\rangle \mid G \text{ is a graph, } k \in \mathbb{Z}, k \geq 0.\}$$

where there is a clique of at least size k in G.

Question 10.08

Membership in NP - Verifiers

Prove INDSET is in NP.

$$\text{INDSET} = \{\langle G, k\rangle \mid G \text{ is a graph, } k \in \mathbb{Z}, k \geq 0.\}$$

where there is an independent set of at least k in G.

Bibliography

[1] L. Alcock and K. Weber, "Undergraduates' Example Use in Proof Construction: Purposes and Effectiveness," English, *Investigations in mathematics learning*, vol. 3, no. 1, pp. 1–22, 2010, ISSN: 1947-7503.

[2] G. MacGillivray, *Notes for Math 122 Logic and Foundations*, English. [Online]. Available: https://www.math.uvic.ca/faculty/gmacgill/LFNotes/M122Notes.pdf.

[3] B. Karlov, "On Theory of Regular Languages with the Kleene Star Operation," English, *Lobachevskii journal of mathematics*, vol. 41, no. 9, pp. 1660–1665, 2020, ISSN: 1995-0802. DOI: https://doi.org/10.1134/S1995080220090164.

[4] Y. Liu, L. Guo, P. Liu, and J. Tan, "Compressing Regular Expressions' DFA Table by Matrix Decomposition," English, in *Implementation and Application of Automata*, ser. Lecture Notes in Computer Science, Berlin, Heidelberg: Springer Berlin Heidelberg, pp. 282–289, ISBN: 3642180973.

[5] R. Lu and H. Zheng, "Pumping Lemma for Quantum Automata," English, *International journal of theoretical physics*, vol. 43, no. 5, pp. 1191–1217, 2004, ISSN: 0020-7748.

[6] M. Bortin, "A Formalisation of the Cocke-Younger-Kasami Algorithm," *Archive of Formal Proofs*, 2016, https://isa-afp.org/entries/CYK.html, Formal proof development, ISSN: 2150-914x.

[7] S. Jain, Y. S. Ong, and F. Stephan, "Regular patterns, regular languages and context-free languages," English, *Information processing letters*, vol. 110, no. 24, pp. 1114–1119, 2010, ISSN: 0020-0190.

[8] S. Iriyama, T. Miyadera, and M. Ohya, "Note on a universal quantum turing machine," English, *Physics letters. A*, vol. 372, no. 31, pp. 5120–5122, 2008, ISSN: 0375-9601.

[9] I. Kim and G. Mahler, "Quantum chaos in quantum turing machines," English, *Physics letters. A*, vol. 263, no. 4, pp. 268–273, 1999, ISSN: 0375-9601.

[10] M. Margenstern, "Non-erasing turing machines: A new frontier between a decidable halting problem and universality," English, in *LATIN '95: Theoretical Informatics*, G. Goos, J. Hartmanis, J. van Leeuwen, R. Baeza-Yates, E. Goles, and P. V. Poblete, Eds., vol. 911, Berlin, Heidelberg: Springer Berlin Heidelberg, 1995, pp. 386–397, ISBN: 978-3-540-59175-7. DOI: 10.1007/3-540-59175-3_104. [Online]. Available: http://link.springer.com/10.1007/3-540-59175-3_104.

[11] M. Muller, "Strongly universal quantum turing machines and invariance of kolmogorov complexity," English, *IEEE Transactions on Information Theory*, vol. 54, no. 2, pp. 763–780, 2008, ISSN: 0018-9448. DOI: 10.1109/TIT.2007.913263. [Online]. Available: http://ieeexplore.ieee.org/document/4439860/.

[12] V. D. Blondel, E. Jeandel, P. Koiran, and N. Portier, "Decidable and undecidable problems about quantum automata," *SIAM Journal on Computing*, vol. 34, no. 6, pp. 1464–1473, 2005. DOI: 10.1137/S0097539703425861. [Online]. Available: https://doi.org/10.1137/S0097539703425861.

[13] F. N. Abu-Khzam, "An improved exact algorithm for minimum dominating set in chordal graphs," *Information Processing Letters*, vol. 174, p. 106 206, 2022, ISSN: 0020-0190. DOI: https://doi.org/10.1016/j.ipl.2021.106206. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0020019021001216.