Monash University
Faculty of Information Technology
2^{nd} Semester 2022

程序代写代做 CS编程辅导

FIT2014
Tutorial 3
Regular Langu[illegible] Definitions, Finite Automata, Kleene's [illegible]Theorem

There may not be [illegible] cover all these exercises, so please attempt them *before* the tutorial and make [illegible] about the ones you did not manage to solve confidently.

Remember that tu[illegible]t giving answers: you'll get these anyway, on Moodle, after the week of the[illegible]im is to *make you think* about the material covered during the lectures and, in particular, to generate discussion about some of the more interesting and challenging concepts.

The Supplementary Exercises are there for extra practice, and it is recommended that you attempt a selection of these, although you will not have time to cover them during the tutorial.

WeChat: cstutorcs

ASSESSED PREPARATION: Question 6.
You must provide a serious attempt at this entire question at the start of your tutorial (for on-campus classes), or submit it online before your tutorial begins (for online classes).

Assignment Project Exam Help

Email: tutorcs@163.com

1. Write down all the words of length less than or equal to 8 described by the following regular expression.

$(\mathbf{ab} \cup \mathbf{ba})(\mathbf{aa} \cup \mathbf{bb}) \mathbf{ab}$

QQ: 749389476

2. For each of the following languages construct a regular expression defining each of the following languages over the alphabet $\{a, b\}$.

https://tutorcs.com

(i) All words that contain exactly two letters.

(ii) All words that contain exactly two *b's* or exactly three *b's*, not more.

(iii) All strings that end in a double letter.

(iv) All strings that do not end in a double letter.

(v) All strings that have exactly one double letter in them.

(vi) All strings that have an even length.

(vii) All strings in which the letter b is never tripled. This means that no word contain the string bbb.

3. This question is inspired by the abstract game Zendo[1][2]

One round of the game has the following structure.

One player is known as the Master. When playing in the tutorial class, your Tutor will play this role to begin with. The Master does the following:

---

[1]designed by Kory Heath (`http://www.koryheath.com/zendo/`), published by Looney Labs (`https://www.looneylabs.com/games/zendo`) on New Year's Eve, 1999

[2]Thanks to FIT2014 tutor Ben Jones for the idea for this question.

程序代写代做 CS编程辅导

1. devise a regular expression, which we'll denote by R, and keep it secret;
2. devise a string x which matches R, and another string y which does not match R;
3. reveal x and y t[illegible]icating which matches R and which does not.

The other players are [illegible] They do the following.

1. Each Student d[illegible]als it to everyone;
2. The Master tell[illegible]h student's string z matches R or not (but does not reveal R).
3. The Master indi[illegible] matches R (but does not reveal R).
4. A Student — or group of Students working together — may guess what R is. Let S be the regular expression they guess.

WeChat: cstutorcs

5. If the guess S is right ($S = R$), then the Student wins this round, and the round ends. if the guess is wrong, then the Master devises and reveals to everyone a new string w, different from all previously revealed strings, on which R and S disagree (i.e., w matches R but not S, or matches S but not R).

Assignment Project Exam Help

6. Go back to Step 1.

Try this activity with a group of your fellow students. A group size of up to half-a-dozen should work ok.

Email: tutorcs@163.com

Some questions to consider as you do so:

- What sorts of regular expressions do you find work well for the Master?
- What sorts of regular expressions are easiest for the Students to guess?

QQ: 749389476

- What modifications do you need to make to the rules, to make the game work better?

4. Consider the recursive definition of regular expressions given in lectures.

https://tutorcs.com

Suppose we dropped subparts (i) and (ii) of part 3. of the definition (so we can't group or concatenate any more), but kept all other parts of the definition. What "regular expressions" would we get? What "regular languages" would result?

Challenge: Think about the effect of dropping other parts of the definition. Is any part of the definition redundant? If so, which one, and why? For any *essential* part of the definition, find a regular expression which would no longer satisfy the definition if that part were dropped.

5. Build a Finite Automaton that accepts only those words that do not end in *ba*.

6. [3]

You are in a maze represented by a rectangular grid. One cell is the start cell, another is the cell you want to get to. Some pairs of adjacent cells have a wall between them, preventing you from moving directly from one to the other. There is at least one path from the start to the end. Suppose that the characters U, D, L, R represent moving up, down, left and right by one grid cell. A string of these characters represents a sequence of movements through the maze, but is only valid if it does not make you bump into a wall. It's ok to visit cells more than once.

Describe an algorithm for converting a given maze into a regular expression over the alphabet {U,D,L,R} which matches exactly those strings which correspond to sequences of moves which solve the maze.

[3] Thanks to FIT2014 tutor Nathan Companez for this question.

程序代写代做 CS编程辅导

Your algorithm may call any algorithm presented in lectures; if you do this, you do not have to list the steps in the algorithm from lectures that you're calling.

7. Find a Nond[illegible]omaton which accepts the languages defined by the following Regular Exp[illegible]

(i) $(a \cup b)^*(aa \cup bb)$

(ii) $((a \cup b)(a \cup b))^*$

(iii) $(aa \cup bb)^*$

(iv) $(ab)^*aa(ba)^*$

(v) $(ab \cup ba)(aa \cup bb)^*(ab \cup ba)$

WeChat: cstutorcs

8. Convert each of the Nondeterministic Finite Automata you found in the previous question into Finite Automata.

Assignment Project Exam Help

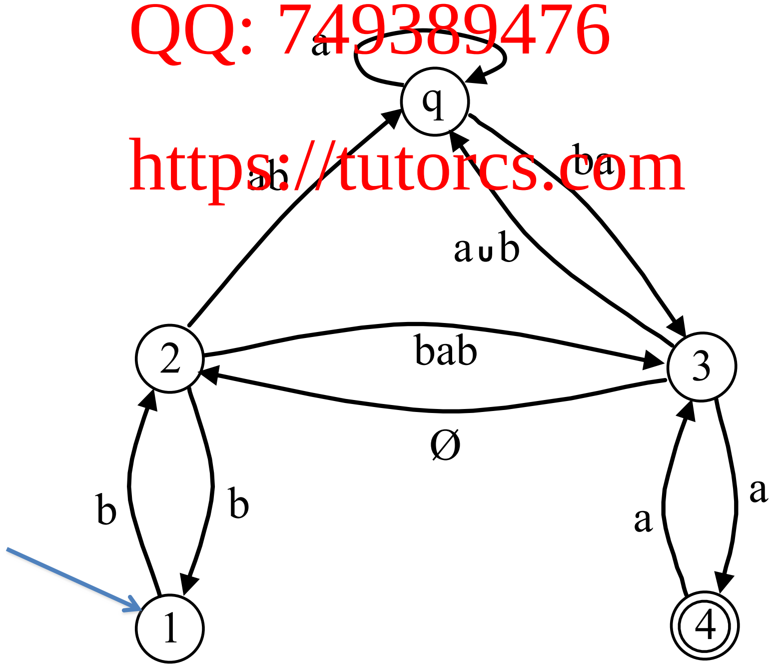
9. *Based on a question from FIT2014 Final Exam, 2nd semester 2015:*

Consider the following Generalised Nondeterministic Finite Automaton (GNFA). Construct an equivalent GNFA with the top state, q, removed.

Email: tutorcs@163.com

QQ: 749389476

https://tutorcs.com



10. Consider the five-state Finite Automaton represented by the following table.

程序代写代做 CS编程辅导

	state	a	b
Start	1	2	3
	2	1	5
	[illegible]	1	4
	[illegible]	2	5
	[illegible]	3	5

Convert this into [illegible] the minimum possible number of states.

WeChat: cstutorcs

Assignment Project Exam Help

Email: tutorcs@163.com

QQ: 749389476

https://tutorcs.com

Supplementary exercises

程序代写代做 CS编程辅导

11. The n-th *harmonic number* H_n is defined by

$$H_n = 1 + \frac{1}{2} + \frac{1}{3} + \cdots + \frac{1}{n}.$$

These numbers have [illegible] computer science.

In this exercise, w[illegible] that $H_n \geq \log_e(n+1)$. (It follows from this that the harmonic numbers inc[illegible] even though the differences between them get vanishingly small so that H_n grow[illegible]y as n increases.)

(i) Inductive basis: prove that $H_1 \geq \log_e(1+1)$.

(ii) *Assume* that $H_n \geq \log_e(n+1)$; this is our *inductive hypothesis*. Now, consider H_{n+1}. Write it recursively, using H_n. Then *use the inductive hypothesis* to obtain $H_{n+1} \geq \ldots\ldots$ (where you fill in the gap). Then use the fact that $\log_e(1+x) \leq x$, and an elementary property of logarithms, to show that $H_{n+1} \geq \log_e(n+2)$.

WeChat: cstutorcs

(iii) In (i) you showed that $H_1 \geq \log_e(1+1)$, and in (ii) you showed that **if** $H_n \geq \log_e(n+1)$ **then** $H_{n+1} \geq \log_e((n+1)+1)$. What can you now conclude, and why?

Assignment Project Exam Help

Advanced afterthoughts:

- The above inequality implies that $H_n \geq \log_e n$, since $\log_e(n+1) \geq \log_e n$. It is instructive to try to prove directly, by induction, that $H_n \geq \log_e n$. You will probably run into a snag. This illustrates that for induction to succeed, you sometimes need to prove something that is *stronger* than what you set out to prove.
- Would your proof work for logarithms to other bases, apart from e? Where in the proof do you use the base e?
- It is known that $H_n \leq (\log_e n) + 1$. Can you prove this?

Email: tutorcs@163.com

QQ: 749389476

12. The *Fibonacci numbers* F_n, where $n \in \mathbb{N}$, are defined by $F_1 = 1$, $F_2 = 1$, and $F_n = F_{n-1} + F_{n-2}$ for $n \geq 3$. The first few numbers in the sequence are $1, 1, 2, 3, 5, 8, 13, \ldots$.

https://tutorcs.com

Prove by induction that the n-th Fibonacci number is given by

$$F_n = \frac{1}{\sqrt{5}}\left(\left(\frac{1+\sqrt{5}}{2}\right)^n - \left(\frac{1-\sqrt{5}}{2}\right)^n\right).$$

To make the algebra easier, give names to the two numbers $(1 \pm \sqrt{5})/2$ and use the fact that they both satisfy the equation $x^2 - x - 1 = 0$.

13. Write down all the words described by the following regular expression.

$(\mathbf{aa} \cup \mathbf{bb})(\mathbf{ba} \cup \mathbf{ab})(\mathbf{aa} \cup \mathbf{bb})$

14. Write down all the words of length 6 described by the following regular expression.

$(\mathbf{aa} \cup \mathbf{bb})*$

15. A programmer used the following regular expression to describe dates where the day, month, and year are separated by "/".

程序代写代做 CS编程辅导

$$([\mathbf{0}-\mathbf{9}][\mathbf{0}-\mathbf{9}] \cup [\mathbf{0}-\mathbf{9}])/([\mathbf{0}-\mathbf{9}][\mathbf{0}-\mathbf{9}])/[\mathbf{0}-\mathbf{9}]$$

Give an example of a valid date not described by this regular expression, and an invalid date described by this regular expres

16. Construct a r scribe times for the twenty-four hour clock, where the hour, minute, and sec ".

17. In the book "*T uage*", by B.W. Kernighan and D.M. Ritchie, a floating point number is define string of numbers possibly containing a decimal point, and an optional expor **E** or **e** followed by a possible signed integer. Construct a regular expression w g point number.

18. In Linux, consult the man page for **grep** or **egrep** to learn the basics of how to use these utilities. (You may find **sed** and **wc** useful for this exercise too.)

WeChat: cstutorcs

Find a file of all English words, which may often be found in Unix/Linux systems in a location like `/usr/dict/words` or `/usr/share/dict/words`, or can easily be found on the web. You could also use one that you constructed in Lab 0.

Write a regular expression to match any vowel.

Assignment Project Exam Help

Write a regular expression to match any consonant.

Write a regular expression to match any word with no vowel or 'y'. Use **grep**, or one of its relatives, to find how many such words there are in your list.

Similarly, determine how many words have no consonants.

Email: tutorcs@163.com

Write a regular expression to match any word whose letters alternate between consonants and vowels. What fraction of English words have this property?

What is the longest run of consonants in an English word? The longest run of vowels?

Can you use this tool to find a list of all English palindromes?

QQ: 749389476

19. This question is about **one-dimensional Go**: see Tute 1, Q13.

Define

$\ell_{B,n}$:= the number of **legal** Go positions on the n-vertex path graph, where vertex n is **Black**.

https://tutorcs.com

$\ell_{W,n}$:= the number of **legal** Go positions on the n-vertex path graph, where vertex n is **White**.

$\ell_{U,n}$:= the number of **legal** Go positions on the n-vertex path graph, where vertex n is **Uncoloured**.

$a_{B,n}$:= the number of **almost legal** Go positions on the n-vertex path graph, where vertex n is **Black**.

$a_{W,n}$:= the number of **almost legal** Go positions on the n-vertex path graph, where vertex n is **White**.

(a) State the values of $\ell_{B,1}$, $\ell_{W,1}$, $\ell_{U,1}$, $a_{B,1}$, and $a_{W,1}$.
(b) Derive recursive expressions for $\ell_{B,n+1}$, $\ell_{W,n+1}$, $\ell_{U,n+1}$, $a_{B,n+1}$, and $a_{W,n+1}$, in terms of $\ell_{B,n}$, $\ell_{W,n}$, $\ell_{U,n}$, $a_{B,n}$, and $a_{W,n}$.
(c) How many of these quantities do you really need to keep, for each n, in order to work out the values for $n+1$?
(d) How do you work out the total number of legal positions on the n-vertex path graph, from $\ell_{B,n}$, $\ell_{W,n}$, $\ell_{U,n}$, $a_{B,n}$, and $a_{W,n}$?

20. This is a follow-up question to Q19, on one-dimensional Go.
(a) Write a regular expression for the set of strings that represent legal positions.
(b) Write a regular expression for the set of strings that represent almost legal positions.

程序代写代做 CS编程辅导

(c) Does there exist a regular expression for the set of strings that represent positions that are neither legal nor almost legal?

21. In the lecture ... ere defined by a recursive definition. Give a recursive definition for arithmet ... se integers, the operators $+$, $-$, $/$, $*$, and brackets, ().

22. The languag ... ver the alphabet $\{a, b\}$ consists of those strings of a and b which are the s ... ackward. Give a recursive definition for the language **PALINDROME**.

23. Construct a Fi ... ccepts only words with b as the second letter.

24. Construct a Finite Automaton that only accepts the words *baa*, *ab*, and *abb* and no other strings longer or shorter.

WeChat: cstutorcs

25. Build a Finite Automaton that accepts only those words that have *more* than four letters.

26. Build a Finite Automaton which only rejects the string *aba*.

Assignment Project Exam Help

27. Build a Finite Automaton that accepts only those words that have an even number of substrings *ab*.

Email: tutorcs@163.com

28. Build a Finite Automaton that accepts precisely the strings of decimal digits that represent positive integers (with no leading 0s) that are multiples of 3.

29. Consider the Finite Automaton represented by the following table.

QQ: 749389476

	state	a	b
Start	1	2	1
Final	2	1	2

https://tutorcs.com

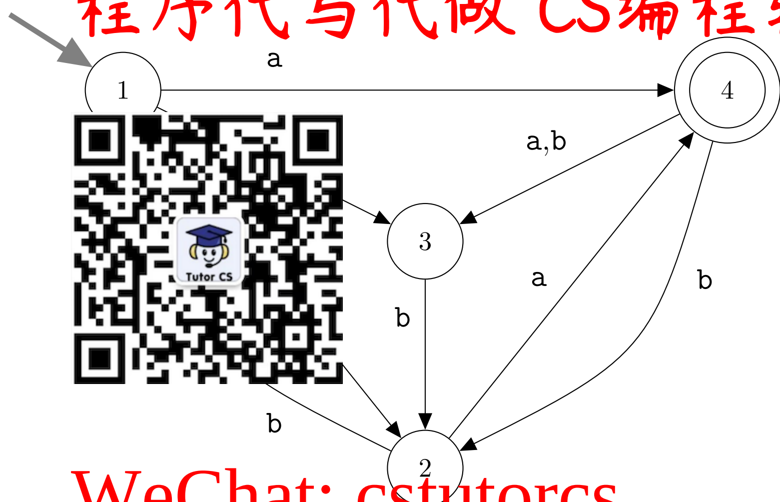
Prove, by induction on the string length, that this FA accepts every string in which a occurs an *odd* number of times.

30.

From FIT2014 Final Exam, 2nd semester 2016:

Consider the following Nondeterministic Finite Automaton (NFA).

程序代写代做 CS编程辅导



WeChat: cstutorcs

(a) What are the possible states that this NFA could be in, after reading the input string `abba`?

Assignment Project Exam Help

(b) Prove, by induction on n, that for all positive integers n, the string $(\texttt{abba})^n$ is accepted by this NFA. (This string is obtained by n repetitions of `abba`.)

Email: tutorcs@163.com

31. Construct a minimum state Finite Automaton that accepts only those strings defined by the regular expression: -?$(N|\ N.|\ N.N|\ .N)$, where $N = [0\text{–}9]^+$.

QQ: 749389476

32. We can prove that two regular expressions are equivalent by showing that their minimum state Finite Automaton are the same, except for state names. Using this technique, show that the following regular expressions are equivalent:

https://tutorcs.com

i. $(a \cup b)^*$

ii. $(a^* \cup b^*)^*$

iii. $((\epsilon \cup a)b^*)^*$

33. For each Finite Automaton you found in Question 8, find the corresponding minimum state Finite Automaton.

34. For each Finite Automaton you found in the previous question, find the corresponding regular expression (using the GNFA approach).