Tema 2 LFA 2024

*citiți vă rog toată cerința înainte să vă apucați de implementat

Se dau următoarele trei probleme, dintre care trebuie să rezolvați doar două (la alegere):

a) Citiți un automat nedeterminist (NFA) din fișier si transformați-l într-un automat determinist (DFA) care acceptă exact acelasi limbaj.
b) Minimizați un DFA (creati un DFA nou, cu numar minim de stari, care acceptă același limbaj ca DFA-ul inițial primit ca input).
c) citiți o expresie regulată și convertiți-o într-un Lambda-NFA. Pentru expresia regulată puteți să simplificați input-ul dacă vreți (de exemplu să puneți paranteze peste tot, să fie mai ușor de parsat).

Cerința c) este mai avansată și cei care aleg să o facă (împreună cu una din primele două cerințe) pot ajunge să aibă nota 12. Cei care fac doar cerințele a și b pot avea maxim nota 10.

Am selectat problema c pentru că seamănă mult cu o problemă dată la ACM ICPC în 2012: 20122013-acmicpc-northeastern-european-regional-contest-neerc-12-en.pdf (codeforces.com) (problema D). La acea problemă trebuie să convertiți două expresii în Lambda-NFA-uri și apoi să vedeți care este cel mai lung string comun acceptat de ambele automate.

Link-uri utile:

Conversie NFA to DFA:

Conversion of NFA to DFA

Conversion of NFA to DFA (Example 1) (youtube.com)
Conversion of NFA to DFA (Example 2) (youtube.com)

Minimizare:

https://en.wikipedia.org/wiki/DFA_minimization
https://www.youtube.com/watch?v=0XaGAkY09Wc&t=781s&ab_channel=NesoAcademy

Expresie regulată:

Regular expression to ∈-NFA - GeeksforGeeks
Thompson's construction - Wikipedia

[Problem of the Day - Two Regular Expressions - Programming 22 - Quora]

Pentru punctaj maxim:

-Programare Orientată pe Obiecte (în C++ sau Python sau orice alt limbaj)

-cu clase

-fără variabile globale

-good coding style:

-nume sugestive la variabile (fără variabile precum a, b, c etc.)

-funcții “mici” (secvențele lungi de cod vor fi sparte în funcții cu puține instrucțiuni și denumiri sugestive)

-cod suficient de clar încât să nu fie nevoie de prea multe comentarii

Recomandare (nu influențează nota voastră)

-folosiți un IDE mai avansat (precum Visual Studio sau CLion sau PyCharm) și învățați să folosiți debugger-ul în loc de afișări la consolă, ajută să depistați mai ușor de ce codul vostru dă crash

Observatie: la tema aceasta nu mai are importanță dacă programul poate citi din fișier doar stări consecutive sau dacă acceptă și stări neconsecutive (cum era în tema anterioară, cu stările 10, 20, 30, 40, etc.)