Uvod u teoriju računarske znanosti

1. blic – pitanja skupljena iz postova

ak. god. 2006/07

1. Okosnica postupka prevodjenja : **prihvacanje** izvornog programa i **generiranje** ciljnog programa.
2. Generiranje gramatike (tipa str 14)
3. Definiranje funkcija pojedinih automata...   
   Prosirenja funkcija automata na argumente...  
   - OKRUŽENJE(q) **= {p| stanje p je stanje q ili -NKA prelazi iz stanja p u stanje q isključivo primjenom -prijelaza}**
4. Mealyev u Mooreov, ako je zadano lambda'(q,b) = \_**b**\_\_
5. ako imaš DKA koji prihvaća ulazni niz 0,1 i imaš stanja p i q, te su funkcije prijelaza.... [sve je standardno zadano] uglavnom, vidiš da za bilo koji ulazni znak (o ili 1) nema prijelaza, ostaje u stanju p. kakav niz DKA prihvaća?
6. za regularni izraz 0 [ona prekrižena nula] definira se jezik **L(0)={}**
7. svojstva stabla : korijen je čvor koji \_\_\_\_\_\_ , a ostali čvorovi \_\_\_\_\_\_ [ugl, sa str10]
8. str 16 --> (2) zadana je lijeva strana relacije i ponuđeno je par odg, treba zaokružiti točan [odnosno taj sa str 16 :)]
9. ako je x pravi prefiks, sufiks ili podniz niza w onda je x \_\_\_\_ w (ponuđeni odgovori su znakovi za jednako, razlicito, slicno i jos neki)
10. automat deterministicki, onda **je delta^(q,a) = delta(q,a)** (ponuđeni odgovori su bili znakovi za jednako, razlicito i jos neki)
11. koji je jezik za reg izraz (1+0)(1+0) **L((1+0)(1+0) = {00,01,10,11}**
12. ako je mealyev automat definiran onako kako i je, onda za moorov vrijedi **Q = Qx**
13. kod DKA se funkcija prijelaza definiria kao: **Q=Qx**
14. Faza analize izvornog programa i faza sinteze ciljnog programa su dvije osnovne faze rada **jezičnog procesora**
15. *Korjen* stabla nema **prethodnika** i vodi do **svih** ostalih čvorova u stablu.
16. nešto kod mijenjanja konačnog automata treba promijeniti:  
    a)simulator konačnog programa  
    **b) tablicu prijelaza** (Želi li se konstruirati konacni automat za neki drugi jezik , potrebno je izgraditi novu tablicu prijelaza, a program simulator nije potrebno mijenjati)
17. ako DKA prima jezik L(M1) U L(M2) onda mora vrijediti da je p iz skupa F i/**ili** q iz skupa F  
    odgovor: ja bih rekla da je ili
18. zadan je neki automat ovako otprilike M = ((p,q),(1,0) ( (q,0)=p, (q,1)=p, (p,1)=p, (p,0)=p) (recimo.........ne znam sad baš točno od znaka do znaka kako je bilo ali tako nekako je) i sad pitanje je kakav niz taj automat prihvaća  
    a) prazan niz  
    b) sve osim praznog niza  
    c) sve i prazan niz (mislim da je tako bilo)  
    d) ništa od navedenog
19. zadan je e-NKA M = ((i,f) S(i,\_\_\_\_\_)=f), i, (f)), tako nekako i treba dopuniti šta fali  
    a) prazan skup  
    b) a iz skupa veliko sigma  
    c) a iz skupa veliko sigma \epsilon(e)  
    d) ......
20. ako iz NKA M radimo DKA M' onda vrijedo da je q0' =\_\_\_\_\_\_\_\_  
    a)q0  
    **b)[q0]**  
    c) O (prazan skup)  
    d)....................
21. ako iz NKA M radimo DKA M' onda vrijedi da je **Q'=**
22. dva automata su istovjetna ako su im istovjetna **POČETNA STANJA**
23. što dobijemo kad maknemo nizu w prefix i sufix? **podniz**
24. dobijete neki regularni izraz i treba napisati koji skup obuhvaća il kaj već-kod mene je izraz bil (0+1)(0+1)--ODGOVOR:**{00,10,01,11}**
25. nekaj s jezičnim procesorom, kaj radi il tak nekaj- **IZVORNI JEZIK PRETVARA U CILJNI**
26. nekaj sa epsilon-NKA i DKA, ako je poč stanje u epsilon-NKA q0 **[q0]**
27. mealy->moore - kolko stanja mora imat moorov ako je zadan mealyjev
28. zadaća jezičnog procesora – **Prevodi zapis algoritma iz izvornog jezika u zapis algoritam u ciljnom jeziku**
29. što sadrži formalni jezik – **Definiciji formalnog jezika pridruzuju se dva matematicka modela, automat i gramatika**
30. Prevođenje zapisa algoritma iz izvornog jezika, u zapis algoritma u ciljnom jeziku obavlja:  
    a) DKA  
    b) NKA  
    **c) Jezicni procesor**  
    d) Nista od navedenog
31. **Sufiks** niza w dobije se odbacivanjem niti jednog, jednog ili vise pocetnih znakova niza w.
32. Ako su zadani DKA M1=(Q1, E, S1, q1, F1) i DKA M2=(Q2, E, S2, q2, F2), onda se DKA M=(Q, E, S, q, F) koji prihvaca regularni jezik L(M) = L(M1) ∩ L(M2) gradi na sljedeći način F = F1xF2, stanje [q, p]eF, gdje je qeF1 **i**/ili qeF2:  
    **a) i**  
    b) ili
33. Regularni izraz (0+1)\* definira jezik:  
    a) (0,1)  
    b) (00,01,10,11)  
    c) (00,1,10,11)  
    **d) Nista od navedenog**  
    e) (e,1,11,111,....111111..)
34. Istovjetnost e-NKA i NKA se dokazuje tvrdnjom e-OKRUZENJE(q0) = S^(q0,x)  
    a) tocno  
    **b) netocno** (e-OKRUZENJE(q0) S^(q0,x))
35. Kod izgradnje NKA iz e-NKA, stanje q0'=  
    a) [q0]  
    **b) q0**  
    c) e-OKRUZENJE(q0)  
    d) Nista od navedenoga
36. Kod vektorskog pristupa programskog ostvarenja konacnog automata najveci problem predstavlja **Memorija**
37. Kod konstrukcije DKA iz NKA, Q' =   
    a) Q X ∆  
    **b) 2^Q**  
    c) Q  
    d) nista od navedenog
38. Jezicni procesor - okosnica rada  
    2. Ekvivalntnost eps-NKA sa NKA  
    3. Pretvoriti Mealyjev u Mooreov  **i**   
    4. Funkcija DeltaKapa u eps-NKA (domena,kodomena)   
    5. Drugi korak algoritma trazenja dostupnih stanja – nadopunjavanje **lista DS se prosiruje skupom stanja {p| za sve a}**  
    6. Automat koji prihvaca samo jezik {eps} - nadopunjavanje  
    7. Izvod istovjetnosti NKA i DKA - da li se navedeni izraz koristi u dokazu?
39. **Duljina niza definira se kao broj znakova niza**
40. Pri pretvorbi Mealy -> Moore, funkcija izlaza Moorea je:  
    **lambda'([qo,b]) = b**
41. Za konacni nederministicki automat vrijedi:  
    delta(q,epsilon) = **{q}**
42. Automat M = {{q,p},{0,1},{ delta(q,0)=p, delta(q,1)=p, delta(p,0)=p, delta(p,1)=p,q,{q,p}} prihvaca  
    **sve nizove, cak i prazan niz**. meni ovo nije bilo ponudjeno, nego sam lijepo na nista od navedenog kliknuo ;)
43. Za autotmat koji prihvaca L(r)={0} treba izgraditi funkciju prijelaza. Zadana su stanja q,p,r. p se prihvaca, q i r se ne prihvacaju.  
    delta(q,0)=p  
    delta(q,1)=r  
    delta(p,0)=r  
    delta(p,1)=r  
    delta(r,0)=r  
    delta(r,1)=r
44. Kod pretvorbe u Mealyev automat:  
    lamba'(qo,a) = delta( lambda(q0) )   
    **Netocno** (lamba'(qo,a) = lambda( delta(q0,a) )
45. DKA M1 prihvaca L(r1), DKA M2 prihvaca L(r2). Za DKA koji prihvaca L(r1)-L(r2) vrijedi ( q iz Q1, p iz Q2 ):  
    **q je element od F1, p nije element od F2**
46. kod implementacije funkcije prijelaza LISTOM, nedostatak je **brzina**  
    (listi treba puno vremena da preko pointera nade novo stanje u koje se prelazi)
47. kako odredujemo da je automat beskonacan:  
    a)preko fje prijelaza NKA  
    **b)preko broja stanja DKA**  
    c)preko fje prijelaza DKA  
    d)izlazom iz Mooreovog automata