

1. Je li jezik Fortran građen za IBM računala?

Da

2. U koje dvije faze je raspodijeljen rad jezičnih procesora?

Analiza izvornog programa i sinteza ciljnog programa.

3. Interpretator naredbe prevodi redosljedom izvođenja.

Točno

4. LEX izgrađuje jezični procesor. Izrazi se zadaju regularnim definicijama. Ciljni jezik mu je C.

Točno

5. Pretraživanje desnog konteksta zadaje se na sljedeći način: r/r'

Točno.

6. Ako je moguće niz znakova grupirati na više različitih načina, onda je leksička analiza

Nejednoznačna.

7. Leksički analizator leksičke jedinice slijedno čita i sve ih zapisuje u tablicu konstanti.

Netočno.

8. Funkcija preslikavanja određuje koliko se naredbi izvornog programa preslikava u koliko naredbi ciljnog programa.

Točno

9. Ako je međukod u grafičkom obliku, redosljed generiranja strojnih naredbi određuje se:

Obilaskom sintaksnog stabla.

10. Dani su regularni izrazi neki poretkom i neki niz. Treba reći kako će LA podijeliti taj niz na leksičke jedinice. U ovom zadatku pazite na redosljed regularnih izraza. Ako niz zadovoljava više od jednog regularnog izraza, niz pripada onoj klasi za koju je regularni izraz naveden najranije u nizu regularnih izraza.

11. Koje produkcije su dio S-gramatike?

a) $\langle S \rangle \rightarrow a b \langle R \rangle$

b) $\langle S \rangle \rightarrow \text{epsilon}$

c) $\langle S \rangle \rightarrow \langle R \rangle a b$

d) $\langle S \rangle \rightarrow c \langle R \rangle$

12. "Najbliži" strojnom kodu su redom (prvo naveden je najbliži):

- a) viši međukod, srednji međukod, niži međukod
- b) niži međukod, srednji međukod, viši međukod

13. Koji su dijelovi sinteze izvornog jezika u izvršivi kod?

- a) leksička analiza
- b) generiranje međukoda
- c) semantička analiza
- d) stojno neovisno optimiranje
- e) sintaksna analiza

14. Leksički analizator stvara tablicu uniformnih znakova i čuva izvorni kod.

Točno.

15. Lex koristi viši programski jezik Java, pritom koristi BNF notaciju. Izlazni jezik je C.

Netočno.

16. Lex koristi viši programski jezik C, izlazni jezik je C.

Točno.

17. Postupak generiranja strojno nezavisnog koda generira:

Kod koji ne ovisi o arhitekturi računala.

18. Strojno zavisno optimiranje ne uzima u obzir posebnosti arhitekture računala na kojem se izvodi ciljni program

Netočno

19. Strojno nezavisno optimiranje ne uzima u obzir posebnosti arhitekture računala na kojem se izvodi ciljni program

Točno

20.

Redni broj pitanja

< 4 >

Broj mogućih točnih odgovora: 1

1 2 3 4 5 6 7 8 9 10

06 m 55 s

Ocjenjivanje provjere

U specifikaciji leksičkog analizatora zadani su regularni izrazi sljedećim redoslijedom: dva / 2, tri / 3 i pet / 5, (a | b | --- | z)*(0 | 1 | --- | 9). Niz znakova i i i i t r i 3 p e t 3 t r i 3 grupira se u leksičke jedinice na sljedeći način:

a)	i i i i t r i 3 # p e t 3 # t r i 3
b)	i i i i # t r i # 3 # p e t # 3 # t r i # 3
c)	i i i i t r i 3 # p e t # 3 # t r i # 3
d)	i i i i t r i # 3 # p e t # 3 # t r i # 3
e)	i i i i # t r i 3 # p e t 3 # t r i 3

21. Redoslijed zapisa znakova u tablici uniformnih znakova ne odgovara redoslijedu leksičkih jedinki u izvornom programu.

Netočno

22. Uobičajeno je da se leksička pravila zadaju primjenom regularnih izraza i regularnih definicija

Točno

23. Nejednoznačnost u leksičkoj analizi može se riješiti analizom lijevog konteksta i analizom desnog konteksta.

Točno

24. Tijekom prevođenja izvornog programa pomoću CO-NO tablica, trenutni operator * i sljedeći operator -> određuju sljedeću akciju generatora ciljnog programa:

- a) spremi
- b) obriši
- c) pomnoži**
- d) dohvati
- e) zbroji

25. Tijekom prevođenja izvornog programa pomoću CO-NO tablica, trenutni operator + i sljedeći operator -> (strela) određuju sljedeću akciju generatora ciljnog programa:

- a) obriši
- b) zbroji**
- c) pomnoži
- d) dohvati
- e) spremi

26. Ona sintaksna analiza primjenom Co-No tablica: Koja akcija se izvesti ako je lijevi operator ; a desni operator -> ?

- a) dohvati**
- b) spremi
- c) izvedi
- d) zbroji

27. Nešto kao: Procesor koji prevodi strojni kod ili procesor strojnog koda je:

- a) računalni procesor**
- b) ...
- c) ...
- d) ...

28. Ako je moguće niz znakova grupirati na više različitih načina, onda je leksička analiza:

- a) Nejednoznačna
- b) neizvediva
- c) nepotrebna
- d) jednoznačna

29. Tijekom prevođenja izvornog programa pomoću CO-NO tablica , trenutni operator + i sljedeći operator -> (strelica) određuju sljedeću akciju generatora ciljnog programa:

- a) obriši
- b) zbroji
- c) pomnoži
- d) dohvati
- e) spremi

30. Leksička pravila definiraju se pomoću regularnih izraza i regularnih definicija.

- a) Točno

31. Redosljed prevođenja i izvođenja interpretatora?

Redosljed prevođenja je određen redosljedom izvođenja

31. Interpretator naredbe prevodi redosljedom izvođenja.

Točno

32. LEX izgrađuje jezični procesor. Izrazi se zadaju regularnim definicijama. Ciljni jezik mu je C.

Točno

35. Što će ispisati lex kad se na ulazni niz ABCabc<1234 primjeni regularni izraz

- a) [a-z]+ ,
- b) [0-9]+ ?

Ima nekoliko ponuđenih odgovora. Točni su:

- a) ABC<1234 ,
- b) ABCabc< .

36. U Co-No notaciji/zapisu dolazi ; koje de se naredbe izvršiti?

- a) zbroji
- b) pomnoži
- c) dohvati
- d) spremi
- e) obriši

37. Program koji prevodi iz nekog koda u strojni kod ili neki oblik koji treba preraditi za strojni kod zove se:

- a) Jezični procesor
- b) Jezični predprocesor
- c) Jezični potprocesor
- d) Program punitelj
- e) Program poveziavač

38. Program koji izvodi kod virtualne mašine nešto nešto

- a) Jezični procesor
- b) Jezični predprocesor
- c) Jezični postprocesor
- d) Program punitelj
- e) Program poveziavač

39. Struktura naredbe zadana sintaksnim pravilima je jednoobrazna za sve tipove naredbi, ili višeobrazna

Točno

40. Kako leksički analizator čita ulazni program?

- a) Liniju po liniju
- b) Znak po znak
- c) Riječ po riječ

41. Za produkciju $A \rightarrow bX$ (X je niz završnih i nezavršnih znakova gramatike), u redak koji odgovara nezavršnom znaku A, stupac koji odgovara završnom znaku b upisuju se sljedeće akcije potisnog automata:

- 1) Izvuci
- 2) Pomakni N

Netočno

42. Co-No tablica, lijevi operator ; i desni operator + definiraju:

Dohvati

43. Što je prvi korak kod sinteze ciljnog programa?

Generiranje međukoda

44. Prilagodba zapisa znakova prethodi leksičkoj analizi.

Točno

45. Neko pitanje s agresivnim skupljanjem informacija o arhitekturi i pesimistično gledanje na nešto...

Točno

46. Prazni znakovi se traže: Ako je znak sa lijeve strane u listi praznih znakova, onda nadopuni listu praznih znakova svim znakovima sa desne strane.

Točno

47.

Kompilator prevodi naredbe izvornog programa onim redoslijedom:

- a) Niti jedno od ponuđenog nije točno
- ☒ b) Kojim su zadane u izvornom programu
- c) Kojim se izvode

48. Kvaliteta ciljnog(?) programa se, između ostalog, mjeri brzinom izvođenja:

Točno

49. Neka je w niz znakova izvornog programa, a x i y su nizovi znakova definirani zadanim regularnim izrazima. Niz znakova x , koji je prefiks niza w , jest leksička jedinka ako i samo ako ne POSTOJI drugi prefiks y niza w takav da je on ujedno i prefiks niza x .

*Netočno (ispravna definicija je na str 58)

50. Java ByteCode je međukod stogovnog stroja.

Netočno

51. Friend javio: Ako se u pitanju spominje nešto "agresivno" i "pesimistično", zaokružite "Točno".

52.

Jezični procesor strojnog programa je primjer jezika čiji jezični procesor nije program, već procesor centralno-procesne jedinice računala.

- a) Netočno
- ☒ b) Točno

53. Program slijedno obrađuju: pretprocesor, jezični procesor, postprocesor, program povezič i program punilac.

Točno

54. koja od navedenih prijelaza nije u Q gramatici (Q gram može bit $\langle N \rangle \rightarrow \epsilon$ ili oblika $\langle N \rangle \rightarrow a \langle M \rangle$)

točno je "<R>-> <S>a"

55. Ako odbacivanjem znakova ulaznog niza pročita znak iz skupa SLIJEDI () onda potisni automat odbacuje nezavršni znak s vrha stoga i nastavi parsiranje.

Točno

56. Redoslijed zapisa znakova u tablici uniformnih znakova odgovara redoslijedu leksičkih jedinki u izvornom programu.

Točno

57. Interpretator čita znakove:

kako se program izvodi

58. Kompajler čita znakove:

Kako je program zapisan

59. Q gramatika ne prihvaća.

<S> --> <A>a [kada je prvi znak nezavršni]