Inacica D: (prvih 20 pitanja su teoretska na zaokruzivanje)

1) Koraci postupka analize izvornog programa redom su:

2) Leksicki analizator slijedno cita tekst izvornog programa. (znak po znak, lijinu po liniju...)

3) Osnovne razine medukoda redom od onih blizem strojnom kodu.

4) U specifikaciji lekisckog analizatora zadani su regularni izrazi sljedecim redoslijedom: (a|b|...|z)\*(0|1|...|9), dva/2, tri/3 i pet/5. Kako se niz petpet5pet5pet3 grupira u leksicke jedine?

5) Zadana je Q-gramatika s produkcijama: S->bAS, S->a, A-> epsilon. Odredite skup PRIMIJENI(A->epsilon).

6) Ako je LR stavka potupna, onda je oznaka tocke: (s koje strane)

7) Koje upravljacke tablice koristi parser Pomakni-Reduciraj?

8) Koji se od navedenih postupaka koristi u pretvorbbi LL(1)-gramatika u Q-gramatiku?

9) Do kakvog inace moguceg proturjecja nikada ne moze dovesti grupiranje stanja koja imaju iste LR(0) stavke?

10) Relacija ReduciranZnakom(A,x) vrijedi ako je znak A pocetni nezavrsni znak gramatike, a znak x je:

11) Zadana je L-atributno prijevodna gramatika(indeksi su u zagradama): X(n1,i2,i3) -> Y(n4)Z(i5)V(i6,n7,n8)W(n9) gdje su i2,i3,i5 i i6 izvedena svojstva, a n1,n4,n7,n8 i n9 nasljedna. Nasljedno svojstvo n7 moze se racunati na temelju svojstva:

12) U LL(1) gramatici, za praznu produkiciju A->alfa, relacija PRIMIJENI izracunava se kao:

13) Izvodenje naredbi za koje vrijedi da se vrijednost relacije \_\_\_\_\_\_\_\_ ne mijenja odreduje se zivotni vijek priduzivanje imena.

14) Dio izvornog programa u kojem je vazeca deklaracija naziva se \_\_\_\_\_\_\_ deklaracije.

15) Dva su osnovna nacina trazenja slobodnog segmenta memorijeskog prostora: potraga za prvim slobodnim segmentom dovoljne veleicine i potraga za slobodnim segmentom koji po svojoj velicini \_\_\_\_\_\_\_\_\_ trazene memorije.

16) Strojno nezavisni program virtualnog stroja u strojni program prevodi: (tko)

17) Generiranje cijelnog programa na temelju postfiksnog sustava oznaka: Ako se u medukodu procitao operator, onda generator primjeni akciju:

18) Pri optimiranju procedura medukoda srednje razine koristiti se pretvorba rekurzivnih poziva procedura u:

19) Tijekom analize strukture pri analizi tijeka izvodenja programa, analiza strukture obilazi graf tijeka izovdenja programa, trazi podgrafove uzorake, zamijen ih jednim zamjenskim cvorom i gradi:

20) Analizu pseudonima cine dva dijela:

21) Neki program treba ga proci a)kompilator b)intepreter (Pn-prevodi, In - izvodi, n- broj reda unutra programa)

22) Za dani kod (neka funkcija koja imi ispisi unutra i izvan slicno primjeru iz knjige A[0] i A[1]) objasni sve vrste prijenosa paramtera i prikazi ispise.

23) Izgradi LR(1) parser za produkcije: S->bA, A->Sa, A-> epsilon

24) Dan je dio koda asembler (kod se drzi sintakste iz knjige) potrebno ga je optimizirati prozorom (3 instrukcije) da se postigne minimalan broj naredbi.

25) Sa atributnom prijevodnom gramatiku izgraditi nesto sto racuna aritmeticke izraze (paziti na prioritete).