Charakterizujte křížový překladač – generuje kód pro jiný počítač, než na kterém probíhá překlad

Objasněte pojem silikonový překladač – logické proměnné představují logické signály, výsledkem je logický obvod

Co to jsou formátory textu? Uveďte příklad – slouží k transformaci textu do čitelnější podoby, zvýrazňování syntaxe, sázení pro tisk - TeX

Charakterizujte kaskádní překladač – chceme překládač A do C a existuje z B do C, pak je jednodušší vytvořit překladač z A do B

Porovnejte výhody a nevýhody interpretačních a kompilačních překladačů - :)

Jaké jsou důvody pro použití víceprůchodového překladače - snazime se o 1 pruchod, 1 derivacni stroj vsechny atributy ... takovy zavislosti mezi atrinbuty, ze to nejde; prekladani na nekolik casti a jeden mezijazyk a druhy mezijazyk (treba malo pameti, moc tezky na jeden krok), moznost optimalizace

Zdůvodněte, proč se nepoužívají čistě interpretační překladače - protože by byl běh programu příliš pomalý

Co to jsou generátory překladačů, uveďte příklad - generuje překladač podle zadané gramatiky

Nakreslete schéma překladače kompilačního typu - (zdrojový program) -> lexikální analýza -> (programové symboly) -> syntaktická analýza (derivační strom) -> zpracování sémantiky -> (program ve vnitřní formě) -> optimalizace -> (upravený program ve vnitřní formě) -> generování kódu -> (cílový program)

Jaká vlastnost gramatiky podmiňuje nekonečnost generovaného jazyka? - rekurzivita

Popište gramatikou reálná čísla s desetinnou částí   
S -> +S | -S | A | A.C A -> 0X | 1B | ... |9B B -> e | 0B | A X -> e C -> 0C | 1C | ... | 9C | D D -> 1 | ... | 9

Jaký je nejvyšší možný počet stavů deterministického KA, má-li ekvivalentní nederm. KA 5 stavů - az do mohutnosti mnozin vsech podmnozin ... (2^5)-1

Popište tvar identifikátoru levou lineární gramatikou - na pravy strane pravidla na zacatku pravidla neterminalni symbol

Zapište pravou lineární gramatiku čísla real v semilogaritmickém tvaru - automat (semilogartimicky 2,3E-6)

Zapište s co nejmenším počtem pravidel gramatiku popisující binární čísla s lichým počtem jedniček.   
S -> 1A A -> 0A | 1B | e B -> 0B | 1A

Uveďte obecný tvar překladových pravidel používaných v LEX – regexp { akce1 v C notaci }

Jaký řetězec rozpoznává LEX, je-li překladové pravidlo dáno např. výrazem \+?[0-9][0-9]\*$ - cele cislo na konci radku, muze mit znamenko plus

Jaký řetězec rozpoznává LEX, je-li překladové pravidlo dáno výrazem \\*[1-9]\* - \* a retezec cislis vecetne prazdneho

Jak řeší lexik. analyzátory problém nalezení symbolu v případě, kdy je jeden symbol prefixem jiného? - hleda ten nejdelsi (cele cislo desetinne cislo); může vyžadovat pomoc od syntaktického analyzátoru (fortran)

Popište formálně zásobníkový automat a význam jeho částí  
Uspořádaná sedmice (Q, T, G, δ, q0, z0, F), kde: Q je konečná množina vnitřních stavů, T je konečná vstupní abeceda, G je konečná abeceda zásobníku, (značeno řeckým písmenem delta) je tzv. přechodová funkce, popisující pravidla činnosti automatu (jeho program), je definována jako zobrazení z Q×(T ∪ {ε})×G\* do Q×G\*, q0 je počáteční stav, Z0 popisuje symboly uložené na počátku v zásobníku, F je množina přijímajících stavů, F ⊆ Q.

Popište přechodovou funkci zásobníkového automatu akceptujícího s prázdným zásobníkem, který je ekvivalentní gramatice G [S]: S -> ( S ) | S ( ) | e - na vrcholu zasobniku S, ma 3 moznosti, jak se nahradi, pro vsechny terminalni symbolky, kdyz je na zasobniku a ve vstupu tentyz, tak se dela srovnani

Jaký jazyk popisuje gramatika např. G [S]: S -> ( S ) | S ( ) | e - popsat ho slovne nebo priklad, co tam patri, nejaka predstava, tady je to: jazyk zavorkovych vyrazu, ktere jsou parovane a mohou byt do sebe vnorovane

Jaký jazyk popisuje gramatika např. G [S]: S -> a S a | b S b | e - retezec 'a' a 'b' a za tim zrcadlovej obraz retezce

Navrhněte gramatiku jazyka (např. jehož věty mají tvar w wreverzní, kde wÎ{0,1}\* - to samy jako predtim

Kdy označujeme větu jazyka jako víceznačnou - kdyz pro ni existujou alespon 2 derivacni stromy

Popište princip způsobu zotavení ze syntaktické chyby v překladači PL/0 - fce test(), ktera zpusobi, ze se preskakuje, dokud se nenarazi na mnozinu povolenych symbolu

Jaké vlastnosti musí splňovat jazyk analyzovatelný rekurzivním sestupem - nemi byt levo-rekurzivni a jednolivy pravy strany by meli bejt odlisitelne

Vysvětlete funkci procedury Test(s1,s2: symset; n: integer); v překladači PL/0 - s1, s2 - mnozny symbolu a fce se vola na zacatku, kdy zjistuje, jaky z tech pravidel se ma pouzit a vola se nakonci pravidla, kdy se overuje, jeslti je to spravne zakoncene; s1, s2 - mnozina tech symbolu, ktery maji bejkt na zacatku pravy strany a to druhy je mnozina symbolu follow toho levostranyho symbolu ... ve druhym pripade je to follow a mnozina symbolu ktery k tomu vyznamove patri a nemely by se prekocit

Zapište gram. aritmetického výrazu s op. + , \*, a závorkami (, ). Zapište levou deriv. věty i + i  
E->E+T E->T T->T\*F T->F F->(E) | i E->E+T->T+T->F+T->i+T->i+F->i+i

Popište princip metody rekurzivního sestupu - metoda, kdy se rekurzivne volaji procedury/fce, tkere odpovidaji neterminalnim symbolum

Charakterizujte syntetizované atributy - vyhodnocuji se zdola nahoru (mista, kde jsou ulozney jednotlivy promenny (adresy))

Popište způsob vyhodnocování dědičných atributů. - vyhodnocuji se shora dolu (napr - adresa odkud se daji ukladat mezivysledky)

Popište zásady konstrukce postfixového výrazu z infixového - zachovane poradi operandu, operatory nasleduji za operandy

Zapište posloupnost postfixových instrukcí pro např. a10 = - (x20 + y30)/(x20 - y30) - zalezi na priorite

Zapište např -2\*(x + y) ^ 3 pro případ 1) nejvyšší, 2) nejnižší precedence operátoru unárního minus a) v prefixové, b) v postfixové notaci - zalezi na priorite - a na zaklade toho se to vyhodnocuje

Přeložte do posloupnosti postfix. instrukcí if (A10 < B 20) then C 30 := (A 10 + B 20 ) \* ( A10 - B20 ); - vybrat postfixovy instrukce (treba z PL/0 nebo ADD, AMOL, ...) a proste to napsat :) nesmime zapomenout na if-false-jump

Přeložte do postfixových instrukcí příkaz while x<y do x:= (x+y) / (x-y); je-li x na adrese 100 a y na adrese 101 - skok zpatky na while a preskok az za while

Popište význam částí dynamické adresy (adresové dvojice) - uroven vnoreni a offset

Formulujte podmínku, kterou musí splňovat program, aby statický řetězec výpočtového zásobníku stále splýval s dynamickým řetězcem - statickej ukazuje na staticky nadrazenej zaznam, dynamickej na dynamicky (odkud je to volany) ... podm: podprogram musi bejt deklarovanej v te urovni, kde je volanej ten prikaz (ale muzou bejt v podprogramu i podprogramy)

Jaké informace o proměnných jsou uloženy v tabulce symbolů překladače jazyka pascalského typu - druh identifikatoru (promenna, fce, navesti, typ, trida, ...), kdyz je to promenna, tak jakej je to typ (jednoduchej, strukturovanej), atd ... to se vymysli :) adresa promenne, na jakej hladine s jakym offsetem

Jaká je časová složitost práce s rozptýleně organizovanou tabulkou symbolů v závislosti na počtu symbolů v programu? - u seznamu je kvadraticka (pouzivaji se vicekrat) a tady vznikaji taky seznamy (hask funkce), obecne taky kvadraticka, ale tabulka ma k-casti a zacinam od urcityho mista a proto je kvadraticka narocnost redukovana cislem 'k'

Jaká je závislost časové režie vyhledávání v netříděně uspořádané tabulce symbolů na počtu jmen v tabulce - kvadraticka

Popište způsob vytváření a práce s frekvenčně uspořádanou tabulkou symbolů - kdyz se najde polozka, posune se k zacatku, casto pouzivany jsou vic nahore a driv se na ne narazi, tabulka se prohledava od zacatku dolu

Jakými vlastnostmi jazyka je podmíněno statické přidělování paměti - nesmi tam bejt nic dynamickyho :) (dynamicke promenne, dynamicke typy, ale ani rekurze!! a ani paralelne provadene aktivity - nevime kdy a kolik se toho spusti)

Popište odlišnost zpřístupnění nelokálních proměnných Pascalu a C - v pascalu nasobne vnorovani v C ne, v C pouze globalni a lokalni promenne - 2 hladiny; v Pascalu jich je vic, hladin libovolne mnoho

Uveďte, jaké údaje ukládá překladač v aktivačním záznamu - aktivacni zaznam odpovida pridelenemu mistu v pameti pro nejakou jednotku a musi tam bejt navratova adresa, misto pro usporadani aktivacnich zaznamu (jak se ma vypoctovy zasobnik zmenit, az skonci dynamicky ukazatel), musi tam byt staticky ukazatel (v Pascalu), musi byt misto pro parametry, musi byt misto pro lokalni promenne

Vysvětlete, jakým mechanismem překladač zajišťuje respektování lokality identifikátorů v blokově strukturovaném jazyce - aby nasel driv, to, oc je lokalni nez to, co je globalni, tak si identifikatory dava do zasobniku

Uveďte datové struktury , které jsou použitelné k přidělování paměti pro  
a) rekurzivně volané procedury a funkce, - zasobnik, halda  
b) dynamické proměnné, - halda  
c) dynamické typy, - zasobnik, halda  
d) paralelně proveditelné programové jednotky - kaktusovy zasobnik, halda

Popište způsob a důvod použití displeje. - moznost hnizdeni podprogramu a sestupuje po statickem retezci, takze zavedeme display a primo ukazujeme na platnou hladinu

Popište význam částí dynamické adresy (adresové dvojice) - hladina a offset

Jaká jazyková omezení budou důsledkem přístupu do výpočtového zásobníku pomocí displeje, který je realizován jako tříprvkový vektor adres - omezeni jazyka by byly: uroven vnoreni maximalne 3

K čemu slouží mapovací funkce pole a na jaké části se člení? - slouzi k tomu, abych se dostal na prislusny prvek pole, dve casti - pevna (odnekud umisteny a je to vlastne zacatek pole) a zavisla na konkretnich indexech

Jaké informace jsou předávány při volání podprogramu, je-li formálním parametrem procedura  
a) v případě statického přidělování paměti, - vse napevno, preda se jako parametr adresa zacatku procedury; b) v případě dynamického přidělování paměti - jeste k tomu pridame ukazatel na staticke prostredi ve kterym je definovana procedura (byl na to priklad)

Co je obsahem deskriptoru třídy u OO jazyků - ukazatel na deskriptor rodice, udaje o offsetech jednotlivych datovych polozek a udaj o virtualnich metodach (zpristupnena tabulka virtualnich metod)

Popište mechanismus zpracování statických metod - (v OO prostedi), hleda se metoda v miste, kde se vyskytne jeji volani, kdyz se nenajde, jde se do rodice a porad vejs a vejs az se metoda nekde najde a kdyz ne, tak je to chyba

Popište mechanismus zpracování dynamických metod - v pripade, ze se vola nejaka metoda a volani je soucasti prikazu, kde se vyskytuje jmeno referencni promenny a a jmeno objektu vazane k promenne a v okamziku, kdy se provadi vytvoreni objektu konstruktorem, tak se provede navazani na tabulku virtualnich metod a tim padem se bude ten objekt .... bude objekt pouzivat metody, ktere mu prislusi, referecni promenna muze odkazovat na tridy, ktere ji patri i na tridy, ktere jsou jeji potomci, provede se navazani na tabulku visrtulanich metod

Popište obecně základní cyklus interpretu - jako zakladni cyklus procesoru :) pracuje se s citacem instrukci, ktery se pri provedeni instrukce inkrementuje a podle instrukce se vyvola ake, ktera se ma delat do instrukce STOP :)

S pomocí algoritmu generování z přednášek rozepište zadaný příklad pro generování z čtveřic - ulozeni do strace, zapamatovani pro kumutativni/nekomutativni, vzit se sebou mustr a podle toho to udelat

S pomocí algoritmu generování z přednášek rozepište zadaný příklad pro generování z trojic -- jako to nahore :)

Uveďte příklad víceznačné gramatiky. Víceznačnost dokažte. - neco vymyslet, aby mela dva derivacni stromy :)

Kdy označujeme větu jazyka jako víceznačnou? - uz tam bylo :)

Může pro víceznačnou gramatiku existovat ekvivalentní gramatika jednoznačná? - ano, nekde tu byl priklad

Operátor umocnění je ve Fortranu pravorasociativní. Zapište pravidla pro aritmetický výraz respektující tuto vlastnost - uzavorkovat s vyrazem napravo (??)

Charakterizujte vztah mezi jazyky s LL(0) gramatikou a regulárními jazyky - LL(0) nemusim koukat na vstup a muzu expandovat - to muzu udelat tehdy, kdyz pro kazdy neterminalni symbol mam jednu pravou stranu a ten jazyk neni rekurzivni a tim padem je konecnej a proto je hloupejsi nez regularni :) je to podmnozina regularnich jazyku

Uveďte formální definici LL(1) gramatiky - pro kazde pravidlo a -> alfa nebo beta musi platiti, ze first(alfa) atd.... nekde to je napsamny :)

Zdůvodněte, proč je každá LL(1) gramatika silná - protoze ta silna gramatika je definovana timhle vzoreckem a pro k=1 to plati

Uveďte nutnou a postačující podmínku pro to, aby gramatika byla silná LL(k) -

K čemu slouží úprava gramatiky zvaná "pohlcení terminálu"? Uveďte příklad. -neco nejak pojmenujeme [Ai] - viz cviceni, pridame pravidlo [Ai] se prepise na neco :)

Příklady převodu na LL gramatiku, konstrukci rozkladové tabulky a rozklad zadané věty - priklad

Uveďte příklady algoritmicky nerozhodnutelných problémů z teorie formálních jazyků - jestli existuje LRk rpo libovolny k, jeslti nejaka gramatike je nebo neni viceznacna

Existuje pro libovolnou gramatiku typu 2 algoritmus pro - to je gramaticka bezkontextova (cisluji se od 0)

a) převod na ekvivalentní nelevorekurzivní gramatiku? - ano, zname alg.  
b) � � LL(k) � - muzeme to zkouset, ale neni alg.  
c) � � LR(k) � - muzeme to zkouset, ale neni alg.  
d) výpočet množin LR(0) položek? - ano, dojdeme ke konfliktu a pak muzeme pocitat dalsi

Zdůvodněte proč LR(k) gramatiky popisují obsáhlejší třídu jazyků než LL(k) - nejakej obrazek - kam dohlidneme, ty dnesni trojuhelniky

Porovnejte mohutnosti (souboru) množin LR(0), LALR(k), LR(k) položek - LR(0) == LALR(k) <= LR(k)

Jaké podmínky musí splňovat množiny LR(0) položek, aby gramatika byla SLR(1)? - kdyz bude '.' na konci, musi byt jedina polozka a kdyz bude '.' nekde jinde, tak je to presun

Popište tvar LR(0) položky a význam jejích jednotlivých částí - maji tvar prepisovaciho pravidla, kde je navic tecka a ta to rodeluje na to, co je v zasobniku a to, co je na vstupu

Příklady konstrukce LR(0) a SLR(1) tabulek a rozklad zadané věty - LARL tma nebudou :)

Jakou metodu syntaktické analýzy používá YACC - pouziva LALR

Jakým způsobem řeší YACC konflikty redukce-redukce - moc se s tim nemazli a kdyz tam vznikne konflikt redukce-redukce, tak vezme to mensi a redukce-presun, udela redukci :)