Teorie Sergiu

Curs 1 :

-fazele unui compilator : slide curs

-multe povesti

Curs 2:

-stringul vid (epsilon)

-productiile se scriu cu -> iar derivarea cu =>

- V = s1, s2, s3; V\* = epsilon, s1, s2, s3, s1s2, s2s3 etc

-propozitiile apartin limbajului doar daca sunt formate doar din terminale

-simbolul de start tot timpu apartine setului de nonterminale

-tipuri de gramatici : regulata, context-free, context-sensitive, recursively enumerable

- regulata : ce avem in dreapta trebuie sa fie terminal , sau 1 singur terminal si un non-terminal

- independenta de context : nu conteaza in ce merge atata timp cat in stanga e doar un non-terminal

- dependent de context : in functie de context, X se poate deriva in ceva (ce are in stanga/dreapta)

- ORICE gramatica regulata e si independenta de context !

- fraza simpla, un singur pas de derivare ca sa ajungi la Hi

- ca sa fie fraza, trebuie sa fie in acelasi subarbore

- 2 gramatici echivalente : daca stringurile formate numai din terminale sunt aceleasi

- limbajul unei gramatici : aplic derivari, dar sa ajung sa am doar terminale

Curs 3 :

- limbajul automatului : toate stringurile recunoscute(acceptate) de catre automat ( si trebuie sa se termine intr-o stare finala)

- automat determinist : ca poti trece doar intr-o anumite stare cu un input

-

Curs 4 :

-la gramatica terminalu e pe partea dreapta, la automat e pe partea stanga

Curs 5:

- arbori diferiti pentru acelasi string => gramatica ambigua

Curs 6:

- automatul e mai complicat daca ne uitam la mai multe simboluri inainte

Curs 7:

-crestem k pana cand automatul este determinist si atunci este gramatica LL(k)

Curs 9:

- ca sa aplicam LL trebuie sa eliminam recursivitatea

Curs 10:

-daca pornesti din Z, verifici ca apartine limbajului gramaticii

-daca pornesti din q0, verifici ca apartine limbajului automatului

-