

Announcements:

- This is a math course with systems applications. Prereq: CompSci 201, CompSci 230 or equiv.
- Course web page:
www.cs.duke.edu/courses/spring26/compsci334
Familiarize yourself with all parts of the web page. 
- Flipped class - reading/quizzes BEFORE
- Read Chapter 1 in the Linz/Rodger book for next time.
- Complete the reading quizzes on Canvas before class.
(Due to Drop/add, QZ01-QZ05 turn off Jan. 22 10, 11:45am!!)
- Course bulletin board: Ed Discussion (get to from Canvas)

- **Course participation required!**

What will we do in Compsci 334?

Questions

- Can you write a program to determine if a string is an integer?

9998.89

8abab

789342

yes!

- Can you do this if your machine had no additional memory other than the program? (can't store any values and look at them again)

yes!

- Can you write a program to determine if the following are correct arithmetic expressions?

$$((34 + 7 * (18/6)))$$

$$(((((((a + b) + c) * d(e + f))))))$$

yes!

- Can you do this if your machine had no additional memory other than the program?

No, need memory
need a stack
or something

- Can you write a program to determine the value of the following expression?

$$((34 + 7 * (18/6)))$$

yes

- Can you write a program to determine if a file is a valid Java program?

yes, syntax

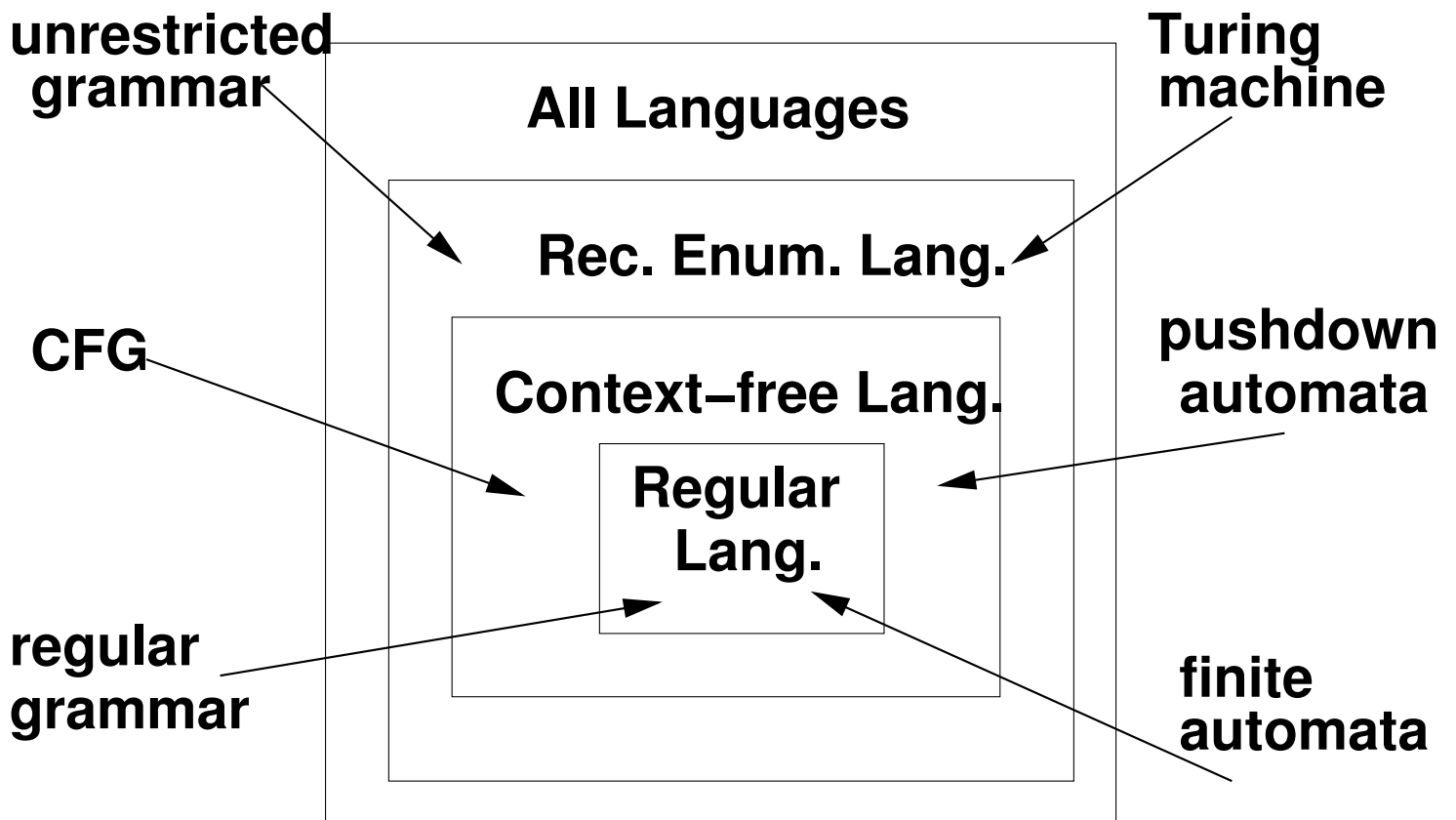
- Can you write a program to determine if a Java program given as input will ever halt?

no! hard problem

Language Hierarchy

Grammars

Automata



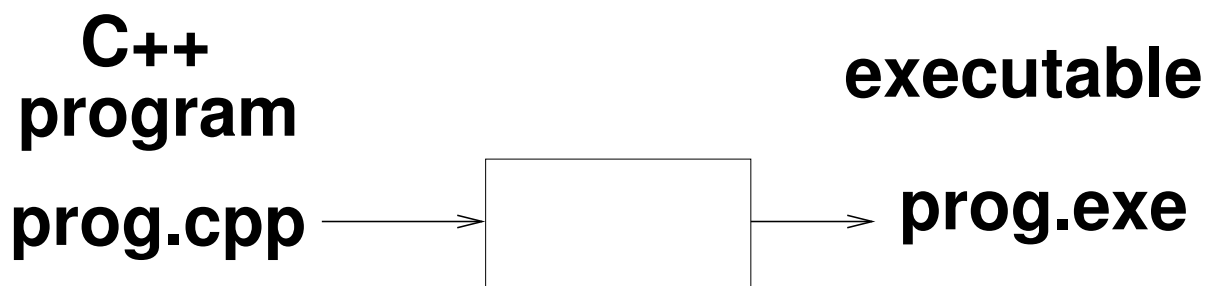
Power of Machines

automata	Can do?	Can't do?
FA (no memory)	recognize integers	recognize arithmetic expression
PDA (stack)	arithmetic expressions	evaluate expr
TM (infinite)	compute expr	decide if halts

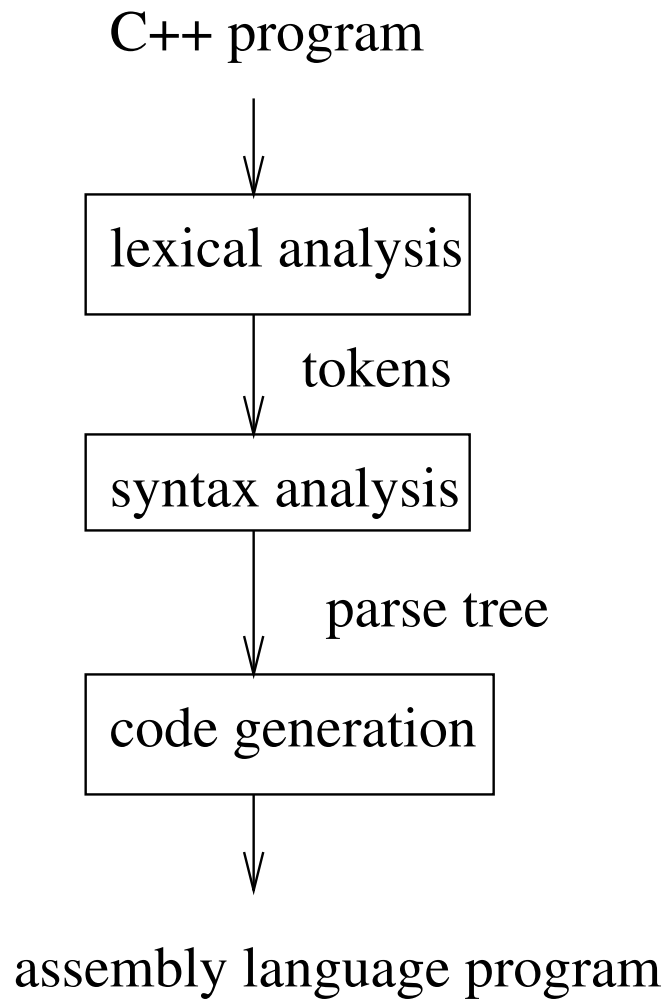
Application

Compiler

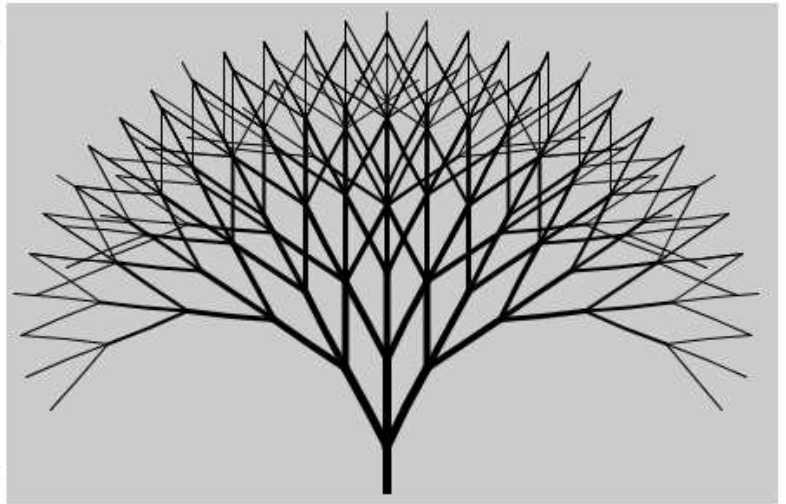
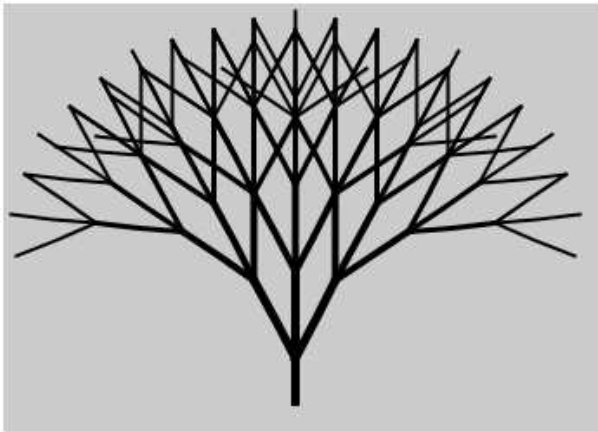
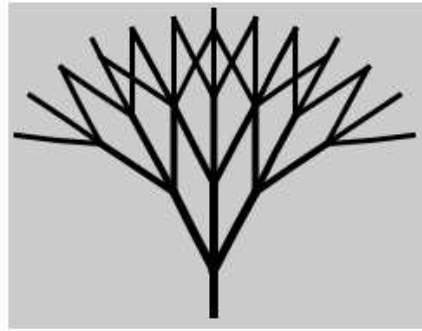
- Our focus - Question: Given a program in some language (say Java or C++) - is it valid?
- Question: language L, program P - is P valid?
- Other things to consider, how is the compilation process different for different programming languages? (Java vs C++?)



Stages of a Compiler



L-Systems - Model the Growth of Plants



Chapter 1 - Set Theory

A Set is a collection of elements.

$$A = \{1, 4, 6, 8\}, B = \{2, 4, 8\},$$

$$C = \{3, 6, 9, 12, \dots\}, D = \{4, 8, 12, 16, \dots\}$$

- (union) $A \cup B = \{1, 2, 4, 6, 8\}$
- (intersection) $A \cap B = \{4, 8\}$
- $C \cap D = \{12, 24, \dots\}$
- (member of) $42 \in C?$ yes
- (subset) $B \subset C?$ No 2 & 4 not in
- $B \cap A \subseteq D?$ yes
- $|B| = 3$
- (product) $A \times B = \{(1, 2), (1, 4), (1, 8), (4, 2), (4, 4), (4, 8), (6, 2), (6, 4), (6, 8), (8, 2), (8, 4), (8, 8)\}$
- $|A \times B| = 12$
- $\emptyset \in B \cap C?$ NO
- (powerset) $2^B = \{\emptyset, \{2\}, \{4\}, \{8\}, \{2, 4\}, \{2, 8\}, \{4, 8\}, \{2, 4, 8\}\}$

Example What are all the subsets of $\{3, 5\}$?

$\emptyset$ $\{3\}$ $\{5\}$ $\{3, 5\}$

How many subsets does a set S have?

$|S|$ number of subsets

0

1

1

2

2

4

3

8

4

16

n

$n+1$

How do you prove? Set S has $2^{|S|}$ subsets.

Technique: Proof by Induction

1. Basis: $P(1)$? *smallest instance*

2. I.H.

Assume $P(n)$ is true for $1, 2, \dots, n$

3. I.S.

Show $P(n+1)$ is true (using I.H.)

Set S has $2^{|S|}$ subsets.

Proof:

1. **Basis:** $|S|=0$ has one element.
 $2^0=1$ ✓

2. **I.H.** Assume $2^{|S|}$ is equal to the number of subsets in S for all $|S| \leq n$

3. **I.S.** Show for $|S|=n+1$ that there are 2^{n+1} subsets

Take one element out $T \cup \{a\} = S$
There are 2^n subsets in T
 S has all subsets of T plus a copy of each with a added.
or $2 \times$ number of subsets in T

$$2 \times 2^n = 2^{n+1}$$

✓

Ch. 1: 3 Major Concepts

- languages
- grammars
- automata

Languages

- Σ - set of symbols, alphabet
- string - finite sequence of symbols
- language - set of strings defined over Σ

alphabet Σ

Examples

- $\Sigma = \{0, 1, 2, 3, 4, 5, 6, 7, 8, 9\}$
 $L = \{0, 1, 2, \dots, 12, 13, 14, \dots\}$

$00 \notin L$

- $\Sigma = \{a, b, c\}$
 $L = \{ab, ac, cabb\}$

- $\Sigma = \{a, b\}$
 $L = \{a^n b^n \mid n > 0\} = \{ab, aabb, aaabbb, \dots\}$

Notation

- symbols in alphabet: $a, b, c, d, \dots$
- string names: $u, v, w, \dots$

Definition of concatenation

Let $w = a_1 a_2 \dots a_n$ and $v = b_1 b_2 \dots b_m$

Then $w \circ v$ OR $wv = a_1 a_2 \dots a_n b_1 b_2 \dots b_m$

String Operations

strings: $w=abbc$, $v=ab$, $u=c$

- size of string

$$|w| + |v| = 6$$

- concatenation

$$v^3 = vvv = v \circ v \circ v = ababab$$

- $v^0 = \epsilon$ empty string

- $w^R = cbba$

- $|vv^Rw| = 8$

- $ab \circ \lambda = ab$

Definition

Σ^* concatenate 0 or more

Example

$$\Sigma = \{a, b\}$$

$$\Sigma^* = \{\epsilon, a, b, aa, ab, ba, bb, aaa, \dots\}$$

$$\Sigma^+ = \Sigma^* - \{\epsilon\}$$

Examples

$$\Sigma = \{a, b, c\}, L_1 = \{ab, bc, aba\},$$

$$L_2 = \{c, bc, bcc\}$$

$$\bullet L_1 \cup L_2 = \{ab, bc, aba, bcc\}$$

$$\bullet L_1 \cap L_2 = \{bc\}$$

$$\bullet \overline{L_1} = \Sigma^* - L_1$$

$$\bullet \overline{L_1 \cap L_2} = \Sigma^* - \{bc\}$$

$$\bullet L_1 \circ L_2 = \{xy \mid x \in L_1 \text{ and } y \in L_2\} =$$

$$\{abc, abbc, abbcc, bc, bcb, bcbcc, abac, \\ ababc, ababcc\}$$

Definition

$$L^0 = \{\lambda\}$$

$$L^2 = L \circ L$$

$$L^3 = L \circ L \circ L$$

$$L^* = L^0 \cup L^1 \cup L^2 \cup L^3 \dots$$

$$L^+ = L^1 \cup L^2 \cup L^3 \dots$$

Grammars

Grammar for english

$\langle \text{sentence} \rangle \rightarrow$

$\langle \text{subject} \rangle \langle \text{verb} \rangle \langle \text{d.o.} \rangle$

$\langle \text{subject} \rangle \rightarrow \langle \text{noun} \rangle \mid$

$\langle \text{article} \rangle \langle \text{noun} \rangle$

$\langle \text{verb} \rangle \rightarrow \text{hit} \mid \text{ran} \mid \text{ate}$

$\langle \text{d.o.} \rangle \rightarrow \langle \text{article} \rangle \langle \text{noun} \rangle \mid \langle \text{noun} \rangle$

$\langle \text{noun} \rangle \rightarrow \text{Fritz} \mid \text{ball}$

$\langle \text{article} \rangle \rightarrow \text{the} \mid \text{an} \mid \text{a}$

Examples (derive a sentence)

Fritz hit the ball.

```
<sentence> -> <subject><verb><d.o>
            -> <noun><verb><d.o>
            -> Fritz <verb><d.o.>
            -> Fritz hit <d.o.>
            -> Fritz hit <article><noun>
            -> Fritz hit the <noun>
            -> Fritz hit the ball
```

Can we also derive the sentences?

The ball hit Fritz.

yes

The ball ate the ball

yes

Syntactically correct?

yes

Semantically correct?

no?

Grammar

$G=(V,T,S,P)$ where

- V - variables (or nonterminals) *can be replaced*
- T - terminals *final, not replaced*
- S - start variable ($S \in V$)
- P - productions (rules)
 $x \rightarrow y \quad x \in (V \cup T)^+, y \in (V \cup T)^*$

Definition

$w \Rightarrow z$ w derives z

$w \xRightarrow{*} z$ w derives in 0 or more steps

$w \xRightarrow{+} z$ w derives in 1 or more steps

Definition of Language of a grammar -
 $L(G)$

$G=(V,T,S,P)$

$L(G)=\{w \in T^* \mid S \xRightarrow{*} w\}$

Example

$$G = (\{S\}, \{a, b\}, S, P)$$

$$P = \{S \rightarrow aaS, S \rightarrow b\}$$

$$L(G) = \{b, aab, aaaaab, \dots\}$$

$= \{(aa)^n b \mid n \geq 0\}$ STOPPED

Example

$$L(G) = \{a^n ccb^n \mid n > 0\}$$

$$G = S \Rightarrow aSb \mid accb$$

Example

$$G = (\{S\}, \{a, b\}, S, P)$$

$$P = \{S \rightarrow aSb, S \rightarrow SS, S \rightarrow ab\}$$

Which of these strings

$aabb, abab, abba, babab$ can be generated by this grammar? Show the derivations.

$$L(G) = \{w \in E^* \mid \text{number of } a\text{'s equal number of } b\text{'s and for every } a \text{ there is a } b \text{ to the right of it.}\}$$

Automata

Abstract model of a digital computer

