LING/C SC/PSYC 438/538

Lecture 21

Prof. Sandiway Fong

Today's Topics

- Homework 12 Review
- **Scheduling Announcement!**
- 538 Presentations (instructions: *choose your topic*)
- Beyond regular languages:
  1. $\{a^n b^n \mid n \geq 1\}$, and
  2. $\{1^n \mid n \text{ is prime}\}$
- A formal tool: the **Pumping Lemma**

Scheduling announcement

from Dec 2nd, I'll be away for research talks in Japan

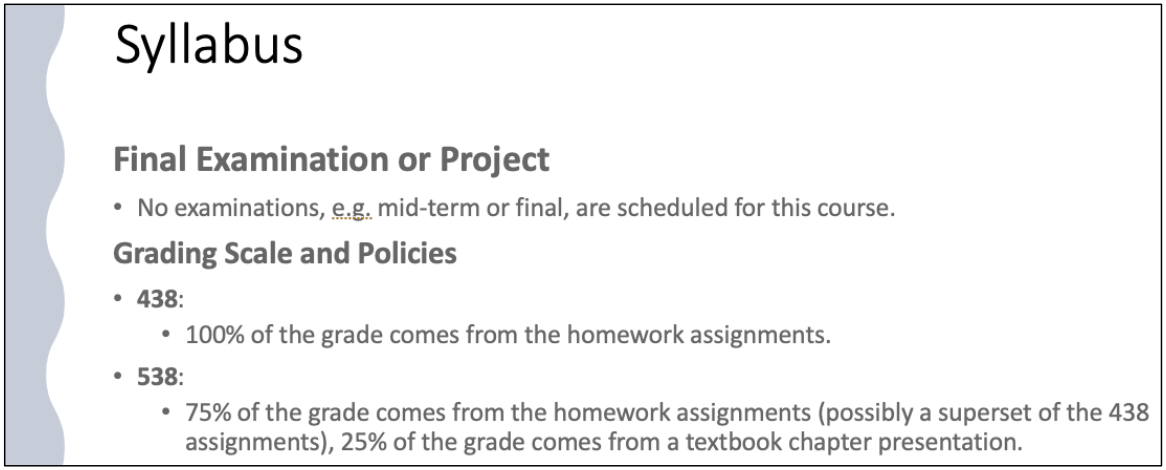


Scheduling announcement

- this means for class scheduling:
  - no lectures in December (3rd and 8th)
  - don't worry, we will still cover the material needed for **581** in Spring
  - last day of 438/538 will be Dec 1st (Nov 26th is *Thanksgiving*)
- we need two classses for **538 Presentations**: Nov 24th and Dec 1st
  - last lecture I can give a homework out is on Nov 19th
  - *for details see updated schedule on class website*

538 Presentation Homework

- 438:
  - no presentation homework
  - only do this if you plan to do the HLT Master's Program (538 is required)
- 538:
  - this is mandatory



Syllabus

Final Examination or Project

- No examinations, e.g. mid-term or final, are scheduled for this course.

Grading Scale and Policies

- **438**:
  - 100% of the grade comes from the homework assignments.
- **538**:
  - 75% of the grade comes from the homework assignments (possibly a superset of the 438 assignments), 25% of the grade comes from a textbook chapter presentation.

538 Homework

- Procedure:
  1. first come, first served
  2. pick a section (*if large enough*) or sections from those offered
     - e.g. *a one page section is probably too short*
     - caution: *the PDF is an incomplete draft*
     - *feel free to supplement the material from other sources*
  3. pick something interesting to you
  4. plan for an 8 min presentation
  5. plan for a couple (or three) of questions
  6. you will be graded on how well you explain the topic in class

538 Homework

- Procedure contd.:
  7. do not simply read the book, use your words: explain, clarify!
  8. make sure you (*quickly*) explain terms introduced

538 Homework

- Procedure contd.:
  - email me your top three choices (ranked) from those offered
  - SUBJECT: 538 Presentation Your Name
  - cite section(s) numbers, title and page numbers in each case
  - choice of presentation dates (ranked)
    - Dec 1st or Nov 24th

538 Homework

- Procedure contd.:
  - once approved (by me), you can go ahead and make slides
    - send them to me, PowerPoint or PDF is fine
    - we will use my laptop for presentations
      - *save (un)plugging time*
    - no "fancy" software
    - you can use the whiteboard too

538 Homework

Use `ed3book.pdf` (*I will email a link to you all*)

- 12 Constituency Grammars 260
- 12.1 Constituency..... 261
- 12.2 Context-Free Grammars..... 261
- 12.3 Some Grammar Rules for English..... 266
- 12.4 Treebanks 273
- 12.5 Grammar Equivalence and Normal Form 278
- 12.6 Lexicalized Grammars..... 279
- 12.7 Summary..... 284

538 Homework

Use ed3book.pdf (*I will email a link to you all*)

- 13 Constituency Parsing 288
- 13.1 Ambiguity.....288
- 13.2 CKY Parsing: A Dynamic Programming Approach 290
- 13.3 Span-Based Neural Constituency Parsing 296
- 13.4 Evaluating Parsers..... 298
- 13.5 Partial Parsing 299
- 13.6 CCG Parsing..... 300
- 13.7 Summary..... 307

538 Homework

Use ed3book.pdf (*I will email a link to you all*)

- 14 Dependency Parsing 310
- 14.1 Dependency Relations..... 311
- 14.2 Dependency Formalisms..... 313
- 14.3 DependencyTreebanks 314
- 14.4 Transition-BasedDependencyParsing 316
- 14.5 Graph-BasedDependencyParsing 325
- 14.6 Evaluation 331
- 14.7 Summary..... 332

538 Homework

Use ed3book.pdf (*I will email a link to you all*)

- 15 Logical Representations of Sentence Meaning 335
- 15.1 Computational Desiderata for Representations 336
- 15.2 Model-Theoretic Semantics.....338
- 15.3 First-Order Logic 341
- 15.4 Event and State Representations..... 348
- 15.5 Description Logics..... 353
- 15.6 Summary..... 359

538 Homework

Use ed3book.pdf (*I will email a link to you all*)

- 17 Information Extraction 363
- 17.1 Relation Extraction 364
- 17.2 Relation Extraction Algorithms.....367
- 17.3 Extracting Times..... 375
- 17.4 Extracting Events and their Times..... 379
- 17.5 Template Filling.....381
- 17.6 Summary..... 383

538 Homework

Use ed3book.pdf (*I will email a link to you all*)

- 18 Word Senses and WordNet 386
- 18.1 Word Senses..... 387
- 18.2 Relations Between Senses..... 389
- 18.3 WordNet: A database of Lexical Relations 391
- 18.4 Word Sense Disambiguation..... 394
- 18.5 Alternate WSD algorithms and Tasks..... 397
- 18.6 Using Thesauruses to Improve Embeddings 400
- 18.7 Word Sense Induction.....400
- 18.8 Summary..... 401

538 Homework

Use ed3book.pdf (*I will email a link to you all*)

- 19 Semantic Role Labeling 405
- 19.1 Semantic Roles 406
- 19.2 Diathesis Alternations..... 407
- 19.3 Semantic Roles: Problems with Thematic Roles 408
- 19.4 The Proposition Bank 409
- 19.5 FrameNet 410
- 19.6 Semantic Role Labeling..... 412
- 19.7 Selectional Restrictions 416
- 19.8 Primitive Decomposition of Predicates 420
- 19.9 Summary..... 422

538 Homework

Use ed3book.pdf (*I will email a link to you all*)

- 20 Lexicons for Sentiment, Affect, and Connotation 425
- 20.1 Defining Emotion 426
- 20.2 Available Sentiment and Affect Lexicons 428
- 20.3 Creating Affect Lexicons by Human Labeling 429
- 20.4 Semi-supervised Induction of Affect Lexicons 431
- 20.5 Supervised Learning of Word Sentiment 434
- 20.6 Using Lexicons for Sentiment Recognition 439
- 20.7 Using Lexicons for Affect Recognition 440
- 20.8 Lexicon-based methods for Entity-Centric Affect 441
- 20.9 Connotation Frames..... 441
- 20.10 Summary..... 443

538 Homework

Use ed3book.pdf (*I will email a link to you all*)

- 21 Coreference Resolution 445
- 21.1 Coreference Phenomena: Linguistic Background 448
- 21.2 Coreference Tasks and Datasets..... 453
- 21.3 Mention Detection..... 454
- 21.4 Architectures for Coreference Algorithms 457
- 21.5 Classifiers using hand-built features. 459
- 21.6 A neural mention-ranking algorithm 461
- 21.7 Evaluation of Coreference Resolution. 464
- 21.8 Winograd Schema problems..... 465
- 21.9 Gender Bias in Coreference.....466
- 21.10 Summary..... 468

538 Homework

Use ed3book.pdf (*I will email a link to you all*)

- 22 Discourse Coherence 472
- 22.1 Coherence Relations..... 474
- 22.2 Discourse Structure Parsing..... 477
- 22.3 Centering and Entity-Based Coherence 481
- 22.4 Representation learning models for local coherence 485
- 22.5 Global Coherence 487
- 22.6 Summary..... 490

538 Homework

Use ed3book.pdf (*I will email a link to you all*)

- 23 Question Answering 494
- 23.1 Information Retrieval 495
- 23.2 IR-based Factoid Question Answering 503
- 23.3 Entity Linking.....507
- 23.4 Knowledge-based Question Answering 511
- 23.5 Using Language Models to do QA 514
- 23.6 Classic QA Models 515
- 23.7 Evaluation of Factoid Answers 518

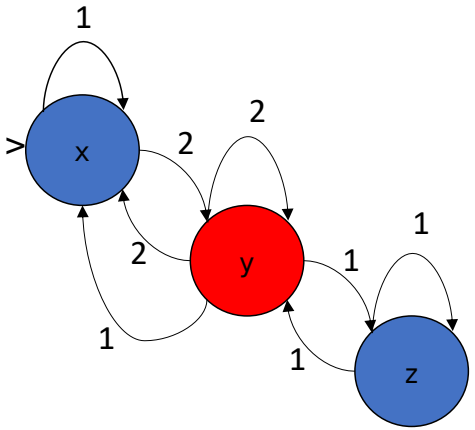
538 Homework

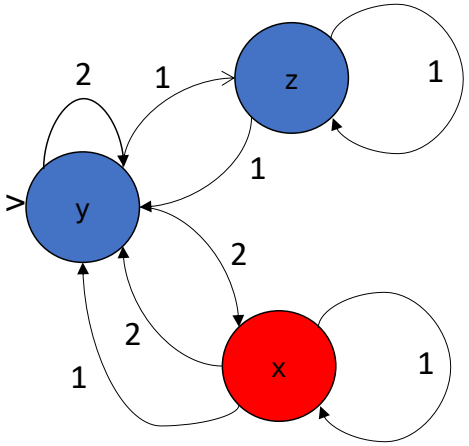
Use ed3book.pdf (*I will email a link to you all*)

- 24 Chatbots & Dialogue Systems 521
- 24.1 Properties of Human Conversation 522
- 24.2 Chatbots 525
- 24.3 GUS: Simple Frame-based Dialogue Systems 533
- 24.4 The Dialogue-State Architecture 537
- 24.5 Evaluating Dialogue Systems 546
- 24.6 Dialogue System Design..... 549
- 24.7 Summary..... 551

Homework 12 Review

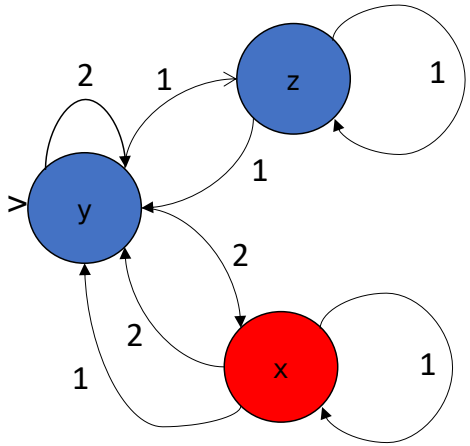
- Q1: $L_R = \{w^R \mid w \in L\}$
- Let's start in state y (*the old end-state*)

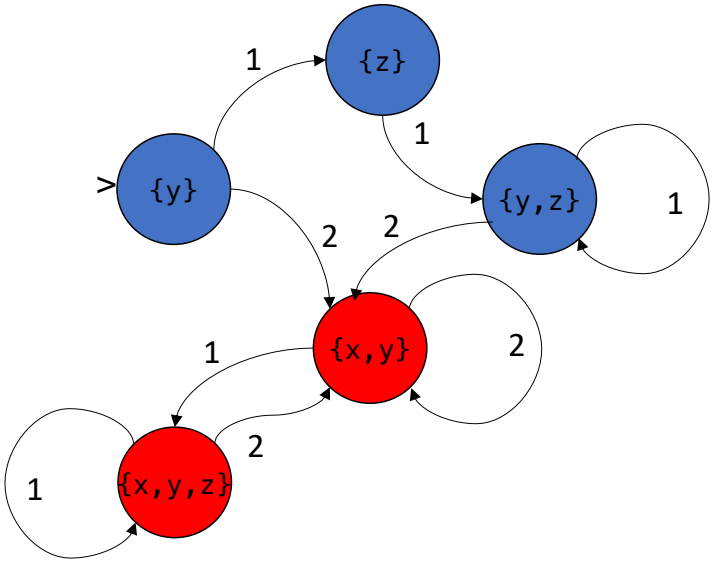




Homework 12 Review

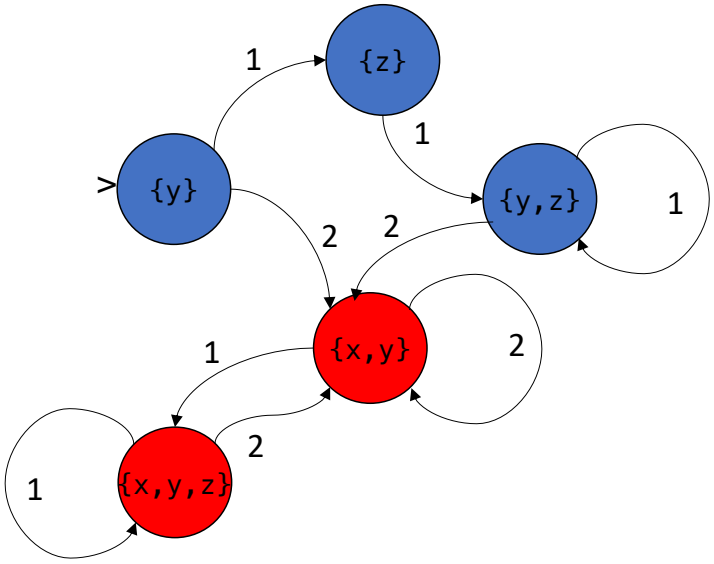
- Q2: convert $L_R = \{w^R \mid w \in L\}$ to a DFSA

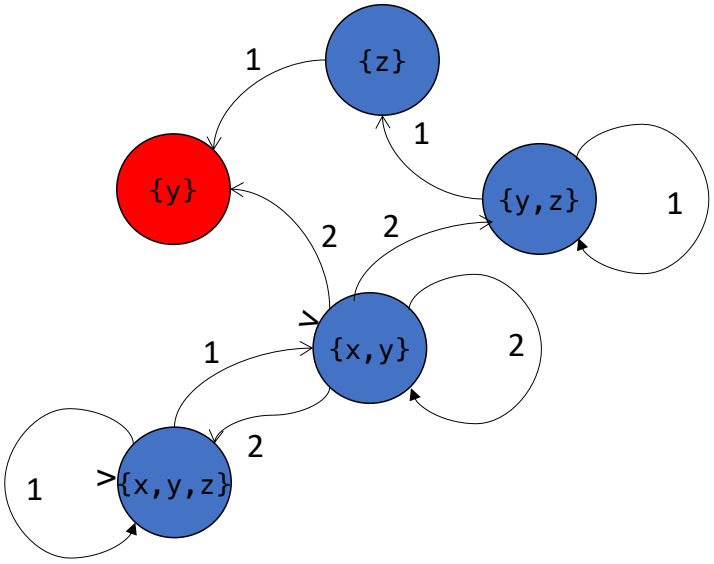




Homework 12 Review

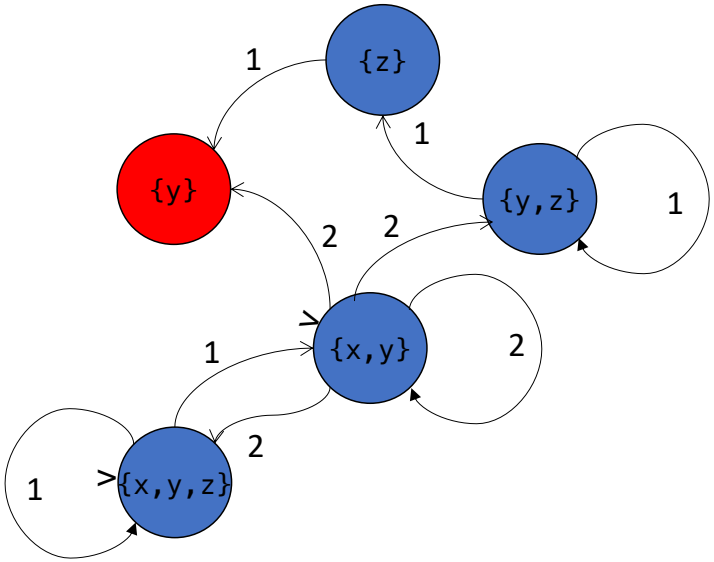
- Q3: $L_{RR} = \{w^R \mid w \in L_R\}$

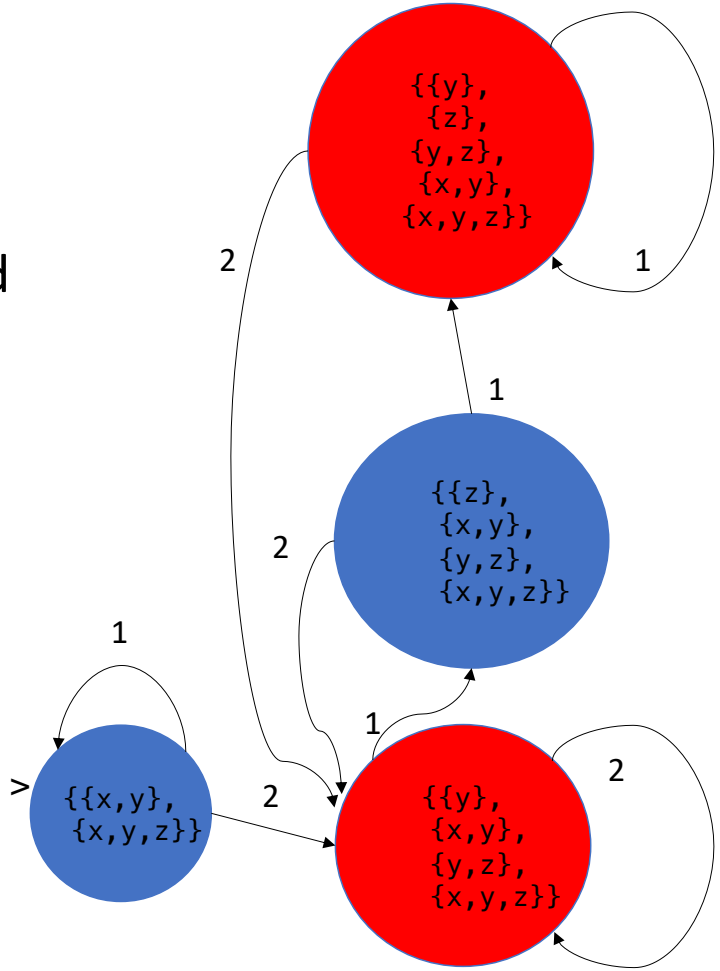




Homework 12 Review

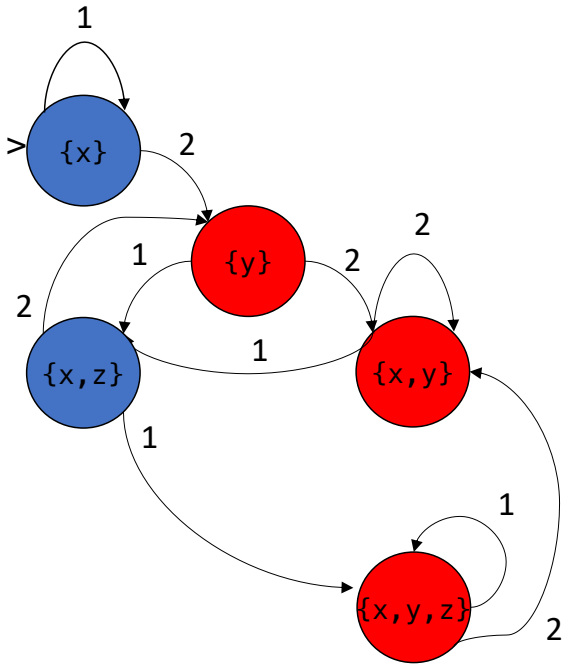
- Q4: $L_{RR} = \{w^R \mid w \in L_R\}$ determinized

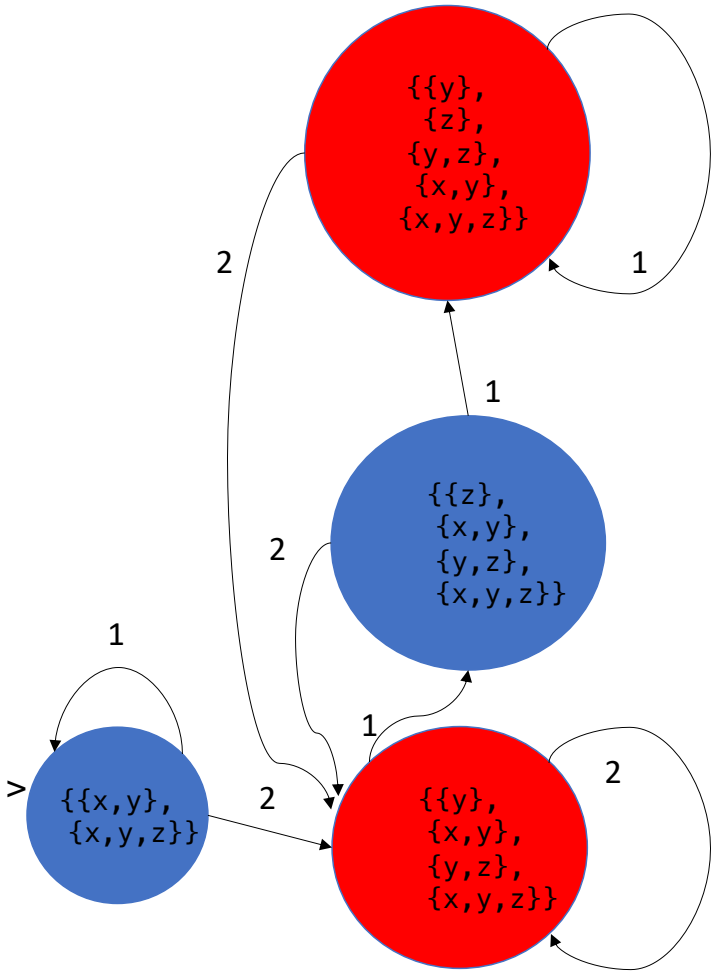




Homework 12 Review

- Compare:





Homework 12 Review

- Transition table:

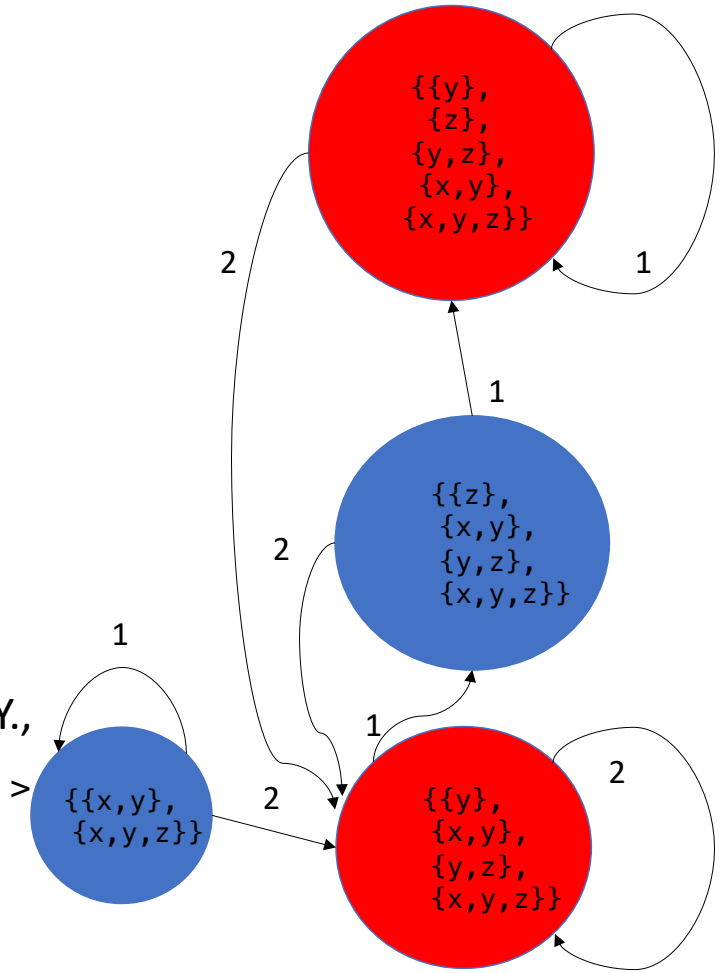
```
%transitiontable = (
    xy_xyz => {1   => "xy_xyz",   2 => "y_xy_yz_xyz"},
    y_xy_yz_xyz => {1 => "z_xy_yz_xyz",  2 => "y_xy_yz_xyz"},
    z_xy_yz_xyz => {1 => "y_z_yz_xy_xyz",  2 => "y_xy_yz_xyz"},
    y_z_yz_xy_xyz => {1 => "y_z_yz_xy_xyz", 2 => "y_xy_yz_xyz"},
);
```

- Run (*compare answer with HW11 Q5*):

```
$ for n in {1..10}; do perl mdfsa.perl $(eval echo
$(printf '{1,2}%.0s' $(seq $n))); done | wc -l
1525
```

State Minimization

- Do you think you could build a deterministic machine for L (= L^{RR}) with fewer states? NOPE
  - Brzozowski, J.A. Canonical regular expressions and minimal state graphs for definite events. In *Proc. Sympos. Math. Theory of Automata (New York, 1962)*, pages 529–561. Polytechnic Press of Polytechnic Inst. of Brooklyn, Brooklyn, N.Y., 1963.



Beyond Regular Languages

- Beyond regular languages
  - $a^n b^n$ = {ab, aabb, aaabbb, aaaabbbb, ... } n≥1
  - is not a regular language

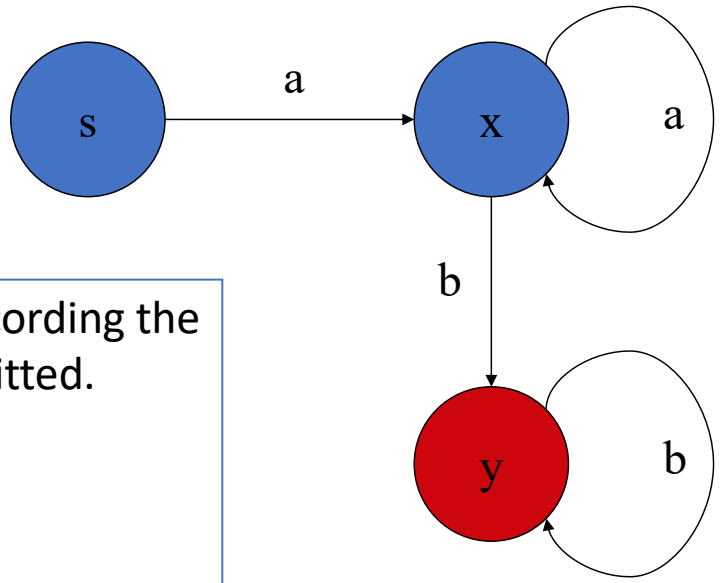
- That means no FSA, regex (or Regular Grammar) can be built for this set

- Informally, let's think about a FSA implementation …

1. We only have a finite number of states to play with …
2. We're only allowed simple free iteration (looping)

Beyond Regular Languages

- $L = a^+b^+$



Having a frequency table recording the number of visits is not permitted.
Not allowed:

```
%freq = ();

… $freq{$state}++;
```

A Formal Tool: The Pumping Lemma

[See also discussion in JM 16.2.1, pages 533–534, *also in many textbooks*]

- Let ***L*** be a regular language,
- then there exists a number $p > 0$
  - where p is a pumping length (*sometimes called a magic number*)

  such that every string w in ***L*** with $|w| \geq p$ can be written in the following form

  $\boldsymbol{w = xyz}$
- with strings x, y and z such that $|xy| \leq p$, $|y| > 0$ and

  $\boldsymbol{xy^i z}$ is in L
- for every integer $i \geq 0$.

BTW: there is also a more complex pumping lemma for Context-Free Languages

A Formal Tool: The Pumping Lemma

Restated:

- For every (*sufficiently long*) string *w* in a regular language
- there is always a way to split the string into three adjacent sections, call them x, y and z, (y nonempty), i.e. w is x followed by y followed by z
- And y can be repeated as many times as we like (or omitted)
- And the modified string is still a member of the language

Essential Point!
To prove a language is non-regular: show that no matter how we split the string, there exist/will be modified strings that can't be in the language.

A Formal Tool: The Pumping Lemma

- Example:
  - show that $a^n b^n$ is not regular
- Proof (by contradiction):
  - pick a sufficiently long string in the language
  - e.g. a..aab..bb (#a's = #b's)
  - Partition it according to ***w = xyz***
  - then show $xy^i z$ is **not** in L
  - i.e. *string does not pump*

A Formal Tool: The Pumping Lemma

aaaa..aabbbb..bb



Case 1: ***w = xyz, y straddles the ab boundary***
what happens when we pump y?

Case 2: ***w = xyz, y is wholly within the a's***
what happens when we pump y?

Case 3: ***w = xyz, y is wholly within the b's***
what happens when we pump y?

A Formal Tool: The Pumping Lemma

- Prime number testing

  prime number testing using Perl's extended "regular expressions"

  - Using unary notation, e.g. 5 = "11111"
  - /^(11+?)**\1**+$/ will match anything that's greater than 1 that's not prime

L = {1^n | n is prime} is not a regular language

A Formal Tool: The Pumping Lemma

1^n = 111..1111..11111 such that n is a prime number



For any split of the string
Pump y such that i = length(x+z), giving y^i

What is the length of string $w=xy^iz$ now?

In x y^{xz} z , how many copies of xz do we have?
Answer is y+1
i.e. pumped number can be factorized into (1+|y|)|xz|

i.e., we can show any prime number can be pumped into a non-prime …

The resulting length is non-prime since it can be factorized

A Formal Tool: The Pumping Lemma

1^n = 111..1111..11111 such that *n* is a prime number



- Illustration of the calculation:

1111 1111 111 (eleven)

1111 1111 1111 1111 1111 1111 1111 1111 111

4 + 4*7 + 3

= 5*7

which isn't prime

- Another look:

1111 111 1111 (re-arrange eleven)

1111 111 1111 1111 1111 1111 111 111 111 111 (make 4 bundles of 4; 4 bundles of 3)

A Formal Tool: The Pumping Lemma

- Another angle to reduce the mystery, let's think in terms of FSA. We know:
  1. we can't control the loops
  2. we are restricted to a finite number of states
  3. assume (without loss of generality) there are no e-transitions

- Suppose there are a total of p states in the machine
- Supose we have a string in the language longer than p
- What can we conclude?

Answer: we must have visited some state(s) more than once!

Also: there must be a loop (or loops) in the machine!

Also: we can repeat or skip that loop and stay inside the language!