



Informatics 2A: Processing Formal and Natural Languages - Introduction

Bonnie Webber
Stuart Anderson



informatics

People

- Lecturers:
 - Bonnie Webber, bonnie@inf.ed.ac.uk, Office Hour, Tues 15-16
 - Stuart Anderson, soa@inf.ed.ac.uk, Office Hour, Tues 13-14
- Teaching Assistants:
 - Srinivasan Chandrasekaran Janarthanam, srinivasancj@gmail.com
 - Katya Alahverdzhieva, K.Alahverdzhieva@sms.ed.ac.uk
- Tutors:
 - Prachya Boonkwan
 - Sharon Goldwater
 - Chris Gorgolewski
 - Tom Kwiatkowski
 - Colin Stirling
- ITO ito@inf.ed.ac.uk:
 - Kendal Reid: kr@inf.ed.ac.uk

Required and Recommended Books

- Your preparations each week involve readings from both these books:
 - Dexter Kozen. **Automata and Computability**. Springer-Verlag, 2000.
 - Dan Jurafsky and James Martin. **Speech and Language Processing. Second Edition**, Prentice-Hall, 2008.
- J.E. Hopcroft, R. Motwani and J.D. Ullman, **Introduction to Automata Theory, Languages and Computation**, Addison-Wesley, 2003, is recommended.
- Steven Bird, Ewan Klein and Edward Loper, **Natural Language Processing with Python**. O'Reilly, 2009 is recommended and available online at <http://www.nltk.org/book>
- These books are essential additions to your personal library of key texts, and will be of use in the years to come.

Required Books: Library Copies

- There are 9 copies of Jurafsky & Martin (2nd edition) in the University libraries:
 - 4 for normal loan in the Main Library
 - 4 for short loan (one week) in the Main Library
 - 1 on RESERVE (3-hour loan) in the Main Library
- There are (at least) 7 copies of Kozen in the University libraries:
 - 1 for normal loan in the Main Library
 - 6 on short loan (one week) in the Main Library
- These are reserve copies but we urge you to purchase your own, through Blackwells or Amazon.

Information Sources

- Informatics 2 web page:
<http://www.inf.ed.ac.uk/teaching/years/ug2/> links to all Inf2 courses and the Informatics 2 Course Guide (the main reference for all Inf2 administration).
- Informatics 2A web page:
<http://www.inf.ed.ac.uk/teaching/courses/inf2a/> links to:
 - Course Descriptor - official spec for the course
 - Teaching Staff - list of people involved in teaching the course
 - Time and Place - this is a list of all possible Inf2a teaching
 - Course Schedule (including slides added after each lecture)
 - Lab Schedule - times of supervised labs and Q&A sessions
 - Tutorials and Labs - see shortly once groups are formed
 - Assignments - available once they have been issued
 - Readings - essential readings outside the course text.

Plagiarism

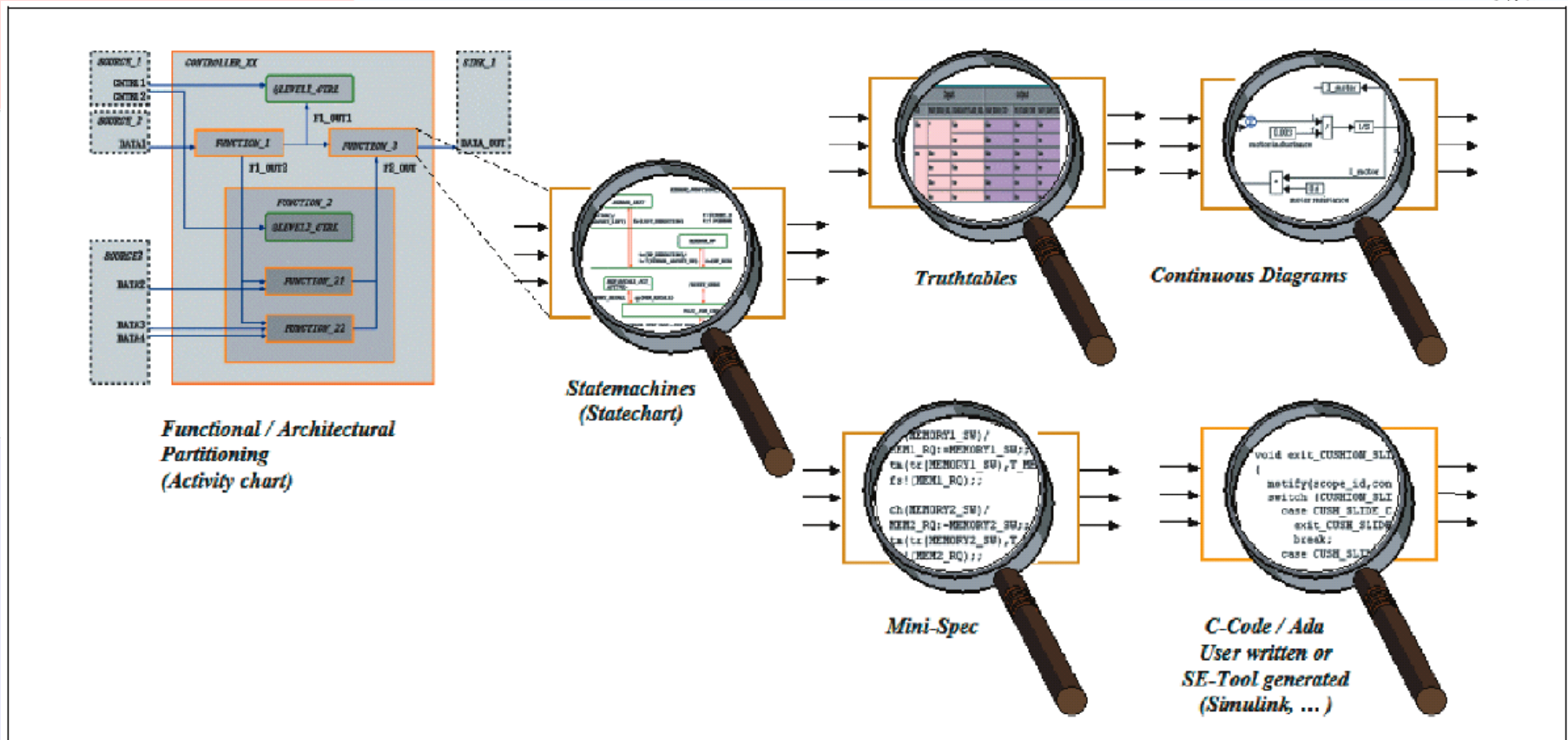
- The University definition of plagiarism is:
 - Plagiarism is the act of copying or including in one's own work, without adequate acknowledgment, intentionally or unintentionally, the work of another, for one's own benefit.
- It is important that you carefully attribute any work that is not your own in all submissions.
 - The University publishes a useful guide on how to avoid plagiarism:
 - [Student Guidance on the Avoidance of Plagiarism](#) [[PDF for printing](#)]
 - Also, please read the school guidelines:
<http://www.inf.ed.ac.uk/admin/ITO/DivisionalGuidelinesPlagiarism.html>
- Part of your education is to develop good habits in attributing the work of others. The above guidance is intended to help you develop this.

Course Overview 1

■ Learning Objectives:

- Demonstrate knowledge of the relationships between languages, grammars and automata, including the Chomsky hierarchy; For example, students will have the capacity to:
 - Construct an appropriate grammar for a given language
 - Construct appropriate automata from grammars and vice versa
 - Use the characteristics of different language classes to demonstrate the feasibility (or otherwise) of building a recogniser for the language.
- Demonstrate understanding of regular languages and finite automata; For example, students will be able to:
 - Design an FSA to recognise a particular language.
 - Demonstrate that a particular language is or is not regular
 - Develop appropriate test sets for finite automata

Why do I need to know about FSMs?

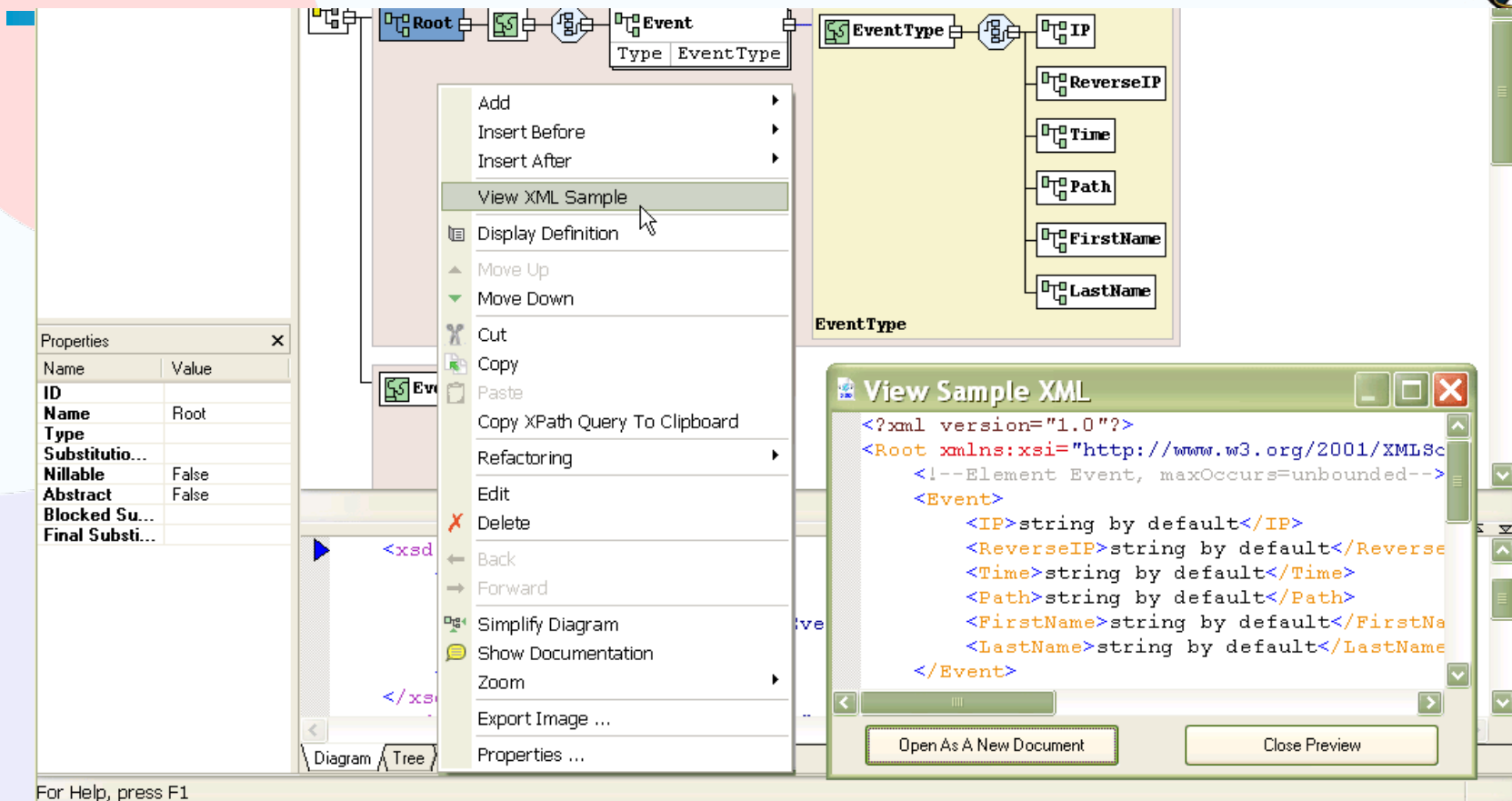


- Basis for many behavioural models
- Commonly used tools like StateMate are based on FSMs (see above)
- The basis of much work on Design and Verification of systems (UML)

Course Overview 2

- Demonstrate understanding of context-free (CF) languages and pushdown automata, and how CF grammars (CFGs) can be used to approximately model a natural language; For example, you should be able to:
 - Design a CFG for a given language (either artificial or natural)
 - Transform a CFG to an equivalent PDA and vice versa
 - Determine whether a given language is or is not CF
 - Be capable of determining whether a given grammar is (un)ambiguous
 - Be capable of providing a compositional interpretation of a given language and be aware of the limitations of the approach.
- Demonstrate knowledge of top-down and bottom-up parsing algorithms for CF languages; For example, you should be able to:
 - Use parsing tools to develop parsers for natural and artificial languages
 - Evaluate the strengths and weaknesses of different parsing strategies and apply that evaluation in choosing an appropriate technique.

Why do I need to know about CFGs?



The screenshot shows an XML editor interface. On the left, a tree view displays the XML structure with nodes for 'Root', 'Event', and 'EventType'. The 'Event' node is selected, and a context menu is open over it, listing various actions such as 'Add', 'Insert Before', 'View XML Sample', 'Display Definition', 'Move Up', 'Move Down', 'Cut', 'Copy', 'Paste', 'Copy XPath Query To Clipboard', 'Refactoring', 'Edit', 'Delete', 'Back', 'Forward', 'Simplify Diagram', 'Show Documentation', 'Zoom', 'Export Image ...', and 'Properties ...'. The 'View XML Sample' option is highlighted. In the foreground, a 'View Sample XML' dialog box is open, displaying the following XML code:

```
<?xml version="1.0"?>
<Root xmlns:xsi="http://www.w3.org/2001/XMLSchema"
  <!-- Element Event, maxOccurs=unbounded-->
  <Event>
    <IP>string by default</IP>
    <ReverseIP>string by default</ReverseIP>
    <Time>string by default</Time>
    <Path>string by default</Path>
    <FirstName>string by default</FirstName>
    <LastName>string by default</LastName>
  </Event>
```

At the bottom of the dialog box, there are two buttons: 'Open As A New Document' and 'Close Preview'. The main editor window also shows a 'Properties' panel on the left with fields for 'Name' (Root), 'Type' (Event), 'Substitution' (False), 'Nillable' (False), 'Abstract' (False), 'Blocked Substitution' (False), and 'Final Substitution' (False). The status bar at the bottom left indicates 'For Help, press F1'.

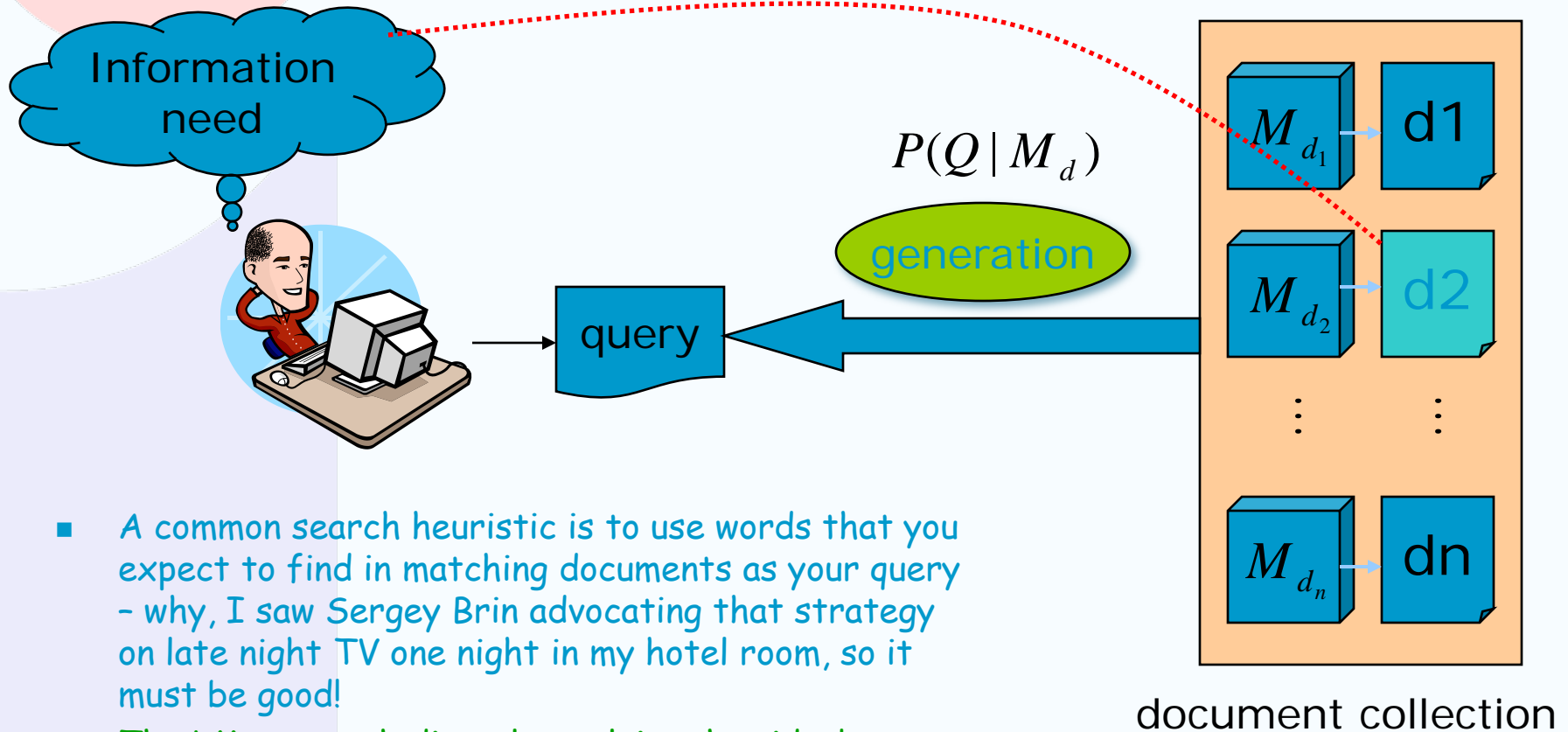
- Underpins the definition of programming languages
- Underpins much of Natural Language Processing
- Semi-structured data - XML

Course Overview 3

- Demonstrate understanding of probabilistic finite state machines (FSMs), including parameter estimation and decoding;
 - You should be able to design simple probabilistic FSMs
- Demonstrate awareness of probabilistic context-free grammars, and associated parsing algorithms; In particular, you should be able to:
 - Use empirical evidence to justify the design of a probabilistic grammar.
 - Demonstrate good and poor design choices in the design of a probabilistic CFG for a given (ambiguous) language.

- Demonstrate knowledge of issues relating to human language processing and to artificial languages. You will study a range of issues including:
 - Ambiguity
 - Compositionality
 - Scope
 - Underspecification

Why do I need ... IR based on Language Model (LM)



- A common search heuristic is to use words that you expect to find in matching documents as your query - why, I saw Sergey Brin advocating that strategy on late night TV one night in my hotel room, so it must be good!
- The LM approach directly exploits that idea!
- Probabilistic languages and grammars underpin LMs

22 September 2009

Inf2A Introductory Lecture

Slide borrowed from
CS276A at Stanford

Course Meetings

- Lectures: Tuesday, Thursday and Friday 16:10-17:00 in Appleton Tower Theatre 3.
- **Please have the week's reading done before attending class.**
- Laboratories: in weeks 2, 3, 5 and 6 in AT, Level 5 "Computer Lab West" on Wednesday @ 11:10; and Friday @ 11:10 and 14:00, and in "Computer Lab North" on Wednesday @ 16:10. The first two labs will focus on Python, the programming language used in the practicals.
- Tutorials: in weeks 2-11, with week 11 for exam revision. All tutorials are in rooms 3.03, 3.05, and 4.07 of Appleton Tower, times are:
 - Tuesday @ 10:00 and 13:05
 - Thursday @ 13:05 and 14:00
 - Friday @ 10:00, 13:05 and 14:00
- **Check for clashes with Inf2c and other classes.**

Course Communication

- Please use eduni.inf.course.inf2a as the first port of call for questions whose answer will be of interest to other students since they will be able to see the question and read the answer. This will also be used to carry course announcements
- On some occasions we will use email to the entire class.
- The Inf 2A homepage will carry announcements of relevance to the whole class
- As announcements will be made at lectures, you are expected to attend lectures.
- It is your responsibility to read news, and mail and keep up to date with the work of the class as reflected in the web page

Examination

- We have no information yet from Registry on the date of the examination.
- Structure:
 - Part 1 consists of compulsory Multiple Choice Questions drawn from across the syllabus.
 - Part 2 consists of longer questions, you will be required to select two questions from a choice of three or four.

Class Representatives

- We need to elect EUSA Class Representatives for this class:
 - http://www.eusa.ed.ac.uk/src/academic/classrep_profile.html
- **Purpose:** As a class rep you are the official representative of your class. You have a positive role to play, by enabling communication and constructive change within your course. Staff within your subject area, the University and the Students' Association value your input, which enables ongoing development and improvement throughout the University.
- You will participate in:
 - Regular meetings with Director of Teaching on management issues for the Informatics teaching areas and on academic liaison.
 - Staff student liaison committee meetings for your courses.
 - School of Informatics Board of Studies and Teaching Committee meetings.
 - Liaison with EUSA on Informatics matters.

Personal Response Systems



- We will use the personal response system, aka "clickers".
- You only need one clicker for all your classes - so if you get one for Inf2C you can use it for Inf2A and any other class.
- Clickers are distributed at the library.
- Get your clicker before the next class on Thursday!

Clicker availability

Where are Clickers available for loan?

1. The main library
2. Somewhere else

Things to do before next meeting

1. Read Jurafsky and Martin (2 ed) Chapter 1.
2. Read Kozen Chapters 1 & 2.
3. Find out about JFLAP:
 1. Visit the JFLAP page: <http://www.jflap.org/>
 2. Read the finite automaton part of the tutorial:
 - <http://www.jflap.org/tutorial/>
 3. Try out the applet:
 - <http://www.cs.duke.edu/csed/jflap/jflaptmp/applet/demo.html>
 4. Use JFLAP to simulate the following machine (drawn for your Inf1a notes):

