

Catalan Structures and Bijections

Student: Stuart Paton
Project supervisor: Dr Anders Claesson
Second Marker: Dr Sergey Kitaev

Say hello and introduce the title of the subject.
Introduce myself.

Aims and Objectives

- Explore various Catalan Structures.
 - Choose between 5 and 10 structures to explore.
- Create bijections between chosen Catalan Structures.
 - Research bijections.
 - Create bijections to correspond to the chosen structures.
- Create a program to visualise Catalan Structures and convert between them using their bijections.
 - In Haskell,
 - Model each structure in a file.
 - Represent them on screen using graphics packages.

Explain the key aims:

Structures so far are:
permutations of S_3 .

Dyck Paths

Young Tableaux (primarily to biject
between dyck path and $Av(132)$ permutations)

Triangulations of an n -gon

Research bijections: Dyck path \rightarrow SSP and
other S_3 permutations are in Classification of
bijections between 321 and 132 avoiding
permutations – Anders Claesson and Sergey
Kitaev.

Bijections over triangulation of an n -gon – still
to be looked at fully.

Program: currently on github \rightarrow

www.github.com/patons02/catalan-structures

Graphics will be done using the Diagrams
package for Haskell.

Background Study

- Learn basics of combinatorics.
 - Binomial theorem
 - Permutations
 - Generating functions
- Catalan structures
 - What are they?
 - Relation to Catalan numbers
 - Proof?
- The following Catalan structures:
 - Dyck Paths
 - Permutations of symmetric group S_3 which avoid any fixed pattern.
 - Young Tableaux
 - Binary Trees
 - Triangulations of an n-gon

Speak of knowledge of combinatorics: how you know about permutations, binomial theorem and generating functions – and how to find catalan numbers from recurrences to generating functions!

Speak about what Catalan structures actually are:

- Formal power series from structures
generating function is the Catalan sequence

- Each Catalan number is of the form:

$$\text{Cat}(n) = C(2n, n)/(n+1)$$

Recurrence:

$$\text{Cat}(0) = 1$$

$$\text{Cat}(n+1) = \sum \text{Cat}(i) \text{Cat}(n-1)$$

Speak of Dyck Paths, permutations, Young Tableaux, binary trees, triangulations of an n-gon (just say what they are and that they're all catalan structures). State that their bijections will be discussed later.

Specification and Requirements

- Model Dyck Paths
- Model permutations that are in the following permutation classes:
 - $Av(123)$, $Av(213)$, $Av(321)$, $Av(231)$, $Av(312)$, $Av(132)$
- Model Young Tableaux
- Model Triangulations of an n -gon
- Visualise the above structures.

Speak about:

Encoding of each structure, and then that it'll be done for each in Haskell and finally that it'll be visualised.

Dyck paths in Haskell
Permutations in Haskell so far
Fact they'll be visualised using
Diagrams.

Project Design

- Each structure is modelled by a recursive formula which will construct and deconstruct the structure.
- Each structure is an instance of the Catalan type class:

```
class Catalan where
  empty :: a
  cons :: a → a → a
  decons :: a → Maybe (a,a)
```
- Dyck paths:
 - Empty: Just an empty list
 - Cons: Put into indecomposable form
 - $D = uDd$ where u = an up-step and d = a down-step
 - Decons: decompose into list of Dyck paths then produce alpha and beta.
- Stack sortable permutations (132-avoiding permutations)
 - Empty: Just an empty string
 - Cons: Make sure it is in the form $\alpha n \beta$ such that all the elements of β are larger than the elements of α and n is the largest element.
 - Decons: Break into alpha and beta disregarding the largest element.

Speak about how to model I will make a recursive mathematical model, and implement in Haskell.
Speak about Catalan typeclass and how each structure is an instance of it (Shown on slide.)

Dyck Path: indecomposable form = $U + \alpha + D$
For the decons part of Dyck paths speak about how it works exactly by getting the height of each element of the Dyck path by their partial sums and how it extracts the alpha and disregards the U and D surrounding alpha, and then also extracts beta.

SSP: Cons: make $\alpha n \beta$ form
Decons: just get alpha and beta, n is just the length of both + 1.

Speak about expansion for other permutations and structures.

Project Design (2)

- Make similar recursive formulae from other structures.
- To get bijections we've simply made a function in order to get bijections for free:

```
bijection :: (Catalan a, Catalan b) => a -> b
bijection w = case decons w of
  Nothing -> empty
  Just (u,v) -> cons (bijection u, bijection v)
```

- We get this since the cons operator, decons operator and empty operators are fully implemented.
-

Speak about getting the bijection for free and how we do, instances of Catalan are defined in the type line so we can then run cons and decons on them both.

Progress Status

- Initial research into Dyck Paths and Permutations of the symmetric class S_3 complete.
 - Proofs that they're Catalan structures
 - Bijections between them
 - Knuth's bijection
 - Knuth-Rotem's bijection
 - Simon-Schmidt's bijection
- Dyck Paths
 - Modelled
 - About to be visualised
- Stack sortable permutations
 - Also known as 132-avoiding permutations
 - Modelled
 - About to be visualised

Speak about overall progress so far:

Mention proofs of each structure and that I've written notes on each of them so far in the form of a paper.

Mention the bijections between Dyck Paths and the permutations of $Sym(3)$.

State how the models of Dyck Paths corresponds with my design.

State how the models of permutations corresponde with my design

Speak a bit about how they're visualised.

Evaluation (1)

- Evaluation will be conducted by analysing the statistics of each structure.
- Statistics will be computed for each permutation.
- Each statistic found for each bijection will be recorded then compared against the other bijections.
- I will start with the following:
 - Dyck Path \rightarrow Stack sortable permutation
 - Stack sortable permutation \rightarrow Dyck path
 - Looking mainly at Knuth's Bijection [1]
- Next – continue for all bijections
- Finally, complete analysis and present best bijection

Speak about the base set of permutation statistics and how they will be referenced against.

Speak about how I will start off finding statistics for each bijection.

Evaluation (2)

- Evaluation will be presented in form of an analysis report.
- Will give a list of all statistics preserved by the structures
- Will analyse their results and present findings on which are the best bijections.

Speak about ways to presenting the findings.

Project Plan

Revised time scale is as follows:

- 11th Jan 2013 – Literature review complete
- 18th Jan 2013 – Analysis of initial structures complete
- 4th Jan 2013 – Model of initial structures complete
- 5th Jan 2013 – Poster presentation complete
- 13th Feb 2013 – Analysis of remaining structures complete
- 20th Feb 2013 – Remaining structures added to program
- 4th March 2013 – Visualisations added to program
- 4th March 2013 – Write up started
- 9th March 2013 – Evaluation complete
- 13th March 2013 – Project submitted to supervisor
- 27th March 2013 – Project submitted for binding
- 29th March 2013 – Project submitted for marking

Speak of how I've revised my plan from original plan.

References

[1] – Anders Claesson and Sergey Kitaev, Classification of bijections between 321- and 132-avoiding permutations, 2008.