

# Student-made Online Discrete Math Drills

## Lightning Talk

Sebastiaan J. C. Joosten  
Department of Computer Science  
Dartmouth College, Hanover, NH 03755  
`Sebastiaan.Joosten@dartmouth.edu`

This paper is the preprint of the article in:  
<http://www.csc.org/publications/journals/NE2021.pdf>

**Introduction.** This paper presents a set of online Discrete Mathematics exercises, and describe how they were made. The motivation is threefold: First, to spread the word about an online environment for doing Discrete Mathematics exercises, in the hope that others might find it useful for their courses. Second, to describe experiential learning in a Functional Programming class. Third, to invite feedback on Discrete Mathematics exercises and suggestions for additional exercises.

**Motivation for online Discrete Mathematics exercises.** A key way in which students learn concepts of Discrete Mathematics is by completing practice exercises. In-person, students can be asked if they practiced, but with the increased prevalence of remote learning, an online environment was created to support the following setup: Online practice exercises are graded homework, students can complete exercises with immediate feedback until they have completed ten in a row correctly. Either full or no points are awarded and no manual grading is involved. The environment is integrated with Canvas.

Discrete Mathematics consists of different topics: set-theory, combinatorics, probability, graphs and numbers. We want basic exercises for all topics. We asked students taking the Functional Programming course to create them.

**Experience with the Functional Programming course.** Students learned Haskell during the first half of the course following the textbook by Richard Bird [1], as during an earlier iteration of this course. For the second half of

the course, students were divided into seven pairs, and each got to pick one out of twelve ‘simple’ assignments. These assignments helped students understand the framework within which the exercises were shown. Next, students got to pick from ‘proof-based’ assignments in fresh groups. These assignments had students generate a derivation and create a Discrete Mathematics exercise based on it. For instance, the Discrete Mathematics student can be asked to put the steps in the right order. Functional Programming students added the exercises through Git pull-requests as a way of submitting their work. On the final day of the course, groups presented their work and they could try each other’s exercises on the departmental server.

The Functional Programming students were enthusiastic about programming something that would actually be used. At the time of writing, student evaluations for the Functional Programming course have not been completed yet, and the Discrete Mathematics course has not taken place yet (this will take place in December and from January to March respectively).

**On the online exercise environment.** The environment consists of generic JavaScript and HTML code to render exercises and feedback, and send responses back to the server. Server-side, the environment is integrated with Canvas. All code is provided to Functional Programming students together with a way to run a single-user exercise server locally. The code is available online at: <https://github.com/sjcjoosten/cs30>  
The application can be tried at: <https://cs.dartmouth.edu/~sjc/cs30/>

## Biography

Sebastiaan Joosten is a Lecturer at Dartmouth. Sebastiaan has a Masters in Discrete Mathematics from Twente University, a PhD in Computer Science from Eindhoven University of Technology (both in the Netherlands). He was previously a PostDoc at the University of Innsbruck (Austria) and an Assistant Professor at Twente University. His research focuses on Automating Correctness, operating at the boundaries of Logic, Mathematics, and Computer Science, and he likes bringing his research into the classroom.

## References

- [1] Richard Bird. *Thinking functionally with Haskell*. Cambridge University Press, 2014.