



## Stichting NIOC en de NIOC kennisbank

Stichting NIOC ([www.nioc.nl](http://www.nioc.nl)) stelt zich conform zijn statuten tot doel: het realiseren van congressen over informatica onderwijs en voorts al hetgeen met een en ander rechtstreeks of zijdelings verband houdt of daartoe bevorderlijk kan zijn, alles in de ruimste zin des woords.

De stichting NIOC neemt de archivering van de resultaten van de congressen voor zijn rekening. De website [www.nioc.nl](http://www.nioc.nl) ontsluit onder "Eerdere congressen" de gearchiveerde websites van eerdere congressen. De vele afzonderlijke congresbijdragen zijn opgenomen in een kennisbank die via dezelfde website onder "NIOC kennisbank" ontsloten wordt.

Op dit moment bevat de NIOC kennisbank alle bijdragen, incl. die van het laatste congres (NIOC2023, gehouden op donderdag 30 maart 2023 jl. en georganiseerd door NHL Stenden Hogeschool). Bij elkaar bijna 1500 bijdragen!

We roepen je op, na het lezen van het document dat door jou is gedownload, de auteur(s) feedback te geven. Dit kan door je te registreren als gebruiker van de NIOC kennisbank. Na registratie krijg je bericht hoe in te loggen op de NIOC kennisbank.

Het eerstvolgende NIOC vindt plaats op donderdag 27 maart 2025 in Zwolle en wordt dan georganiseerd door Hogeschool Windesheim. Kijk op [www.nioc2025.nl](http://www.nioc2025.nl) voor meer informatie.

Wil je op de hoogte blijven van de ontwikkeling rond Stichting NIOC en de NIOC kennisbank, schrijf je dan in op de nieuwsbrief via

[www.nioc.nl/nioc-kennisbank/aanmelden-nieuwsbrief](http://www.nioc.nl/nioc-kennisbank/aanmelden-nieuwsbrief)

Reacties over de NIOC kennisbank en de inhoud daarvan kun je richten aan de beheerder:

R. Smedinga [kennisbank@nioc.nl](mailto:kennisbank@nioc.nl).

Vermeld bij reacties jouw naam en telefoonnummer voor nader contact.

## Testing & Ask-Elle

### *Adding testing to ASK-ELLE – Demonstration*

*Door: Johan Jeuring.*

*Met medewerking van: Bastiaan Heeren, Alex Gerdes (Open Universiteit) en Jurriën Stutterheim (Universiteit Utrecht).*

*Kernwoorden: functioneel programmeren, intelligent tutoring, testen van student-programma's.*

**Adding testing to Ask-Elle: An Interactive Functional Programming Tutor: in this demonstration ASK-ELLE is introduced: a Haskell tutor. ASK-ELLE supports the incremental development of Haskell programs. ASK-ELLE checks that a student follows one of the model solutions provided by a teacher. It can recognise many incomplete variants of the model solutions, it can give hints on how to proceed with solving a programming exercise, and feedback on incomplete student programs, but if a student implements an incorrect solution, or a solution that differs from the model solutions, ASK-ELLE cannot give feedback. For this reason, we have added testing to ASK-ELLE.**

QuickCheck is a property-based testing framework, that tries to generate counterexamples for properties. For each problem in ASK-ELLE, we specify its properties as QuickCheck properties. If we cannot categorize an incomplete student program as part of a model solution, we use QuickCheck to try to generate a counterexample for the program, and report the counterexample to the student. Besides a general demo of ASK-ELLE, the new feature is demonstrated that uses testing to check if incomplete student programs are still correct. ASK-ELLE in action was shown to the attendants followed by discussion how a teacher can configure its behavior.

### **Learning to program**

Learning to program is hard due to misconceptions about the syntax and semantics of a programming language and the difficulty of analysing and creating a model of the problem that can be implemented. Decomposing a complex problem into smaller subproblems requires experience and most compilers give poor error messages and are not helpful.

A programming tutor supports a student when learning how to program by giving hints (in varying level of detail), by showing worked-out solutions and by reporting erroneous steps.

The challenges for programming tutors are to increase their use because they are not widely used. However building a tutor is a substantial amount of work and using a tutor in a course is hard for a teacher: adapting or extending a tutor is often very difficult or even impossible.

Having to specify feedback with each new exercise is often a lot of work.

Preferably, a programming tutor supports easy specification of exercises and automatically derives feedback and hints. The research on ASK-ELLE is aiming to improve these characteristics.

Wilt u reageren op deze presentatie / demonstratie? Neem dan contact op met:

Johan Jeuring; onderzoeker; School of Computer Science; Open Universiteit.

J.T.Jeuring@uu.nl