

<b>Module title</b> Functional Programming				
<b>Module code</b> Tba.	<b>Level</b> Bachelor (B.Sc.)	<b>Hours per week</b> 4	<b>ECTS credits</b> 5	<b>Duration</b> 14 weeks
<b>Module instructor</b> Mairead Meagher Waterford Institute of Technology, Ireland		<b>Lecture type</b> On-line lectures/labs with supporting material.	<b>Prerequisite(s)</b> Introductory programming	<b>Grading</b> Programming assignment, online quiz
<b>Content:</b>				
<ul style="list-style-type: none"> <li>• Introduction to functional programming: why and when should this paradigm be used</li> <li>• Use of a functional language (e.g., Haskell using GHCi, Stack)</li> <li>• Types and classes, functions (lambda expressions)</li> <li>• Recursive functions, including List Comprehensions</li> <li>• Higher-order functions</li> <li>• Declaring Types and Classes</li> <li>• Processing structured data (e.g. trees) via functions</li> <li>• Interactive Programming</li> <li>• Functors, Applicatives and Monads</li> <li>• Current trends in functional programming</li> </ul>				
<b>Upon the end of the module the students will have attained the following subject-matter competencies...</b>				
<ul style="list-style-type: none"> <li>• to apply the functional style of programming to a range of simple problems</li> <li>• to write a simple system using a functional programming language e.g. Haskell which includes interactive programming</li> <li>• to code using functional techniques, e.g. recursion throughout</li> <li>• to use Higher-Order functions when appropriate</li> <li>• to apply functional technique to a specific problem of intermediate complexity and implement the solution</li> <li>• to structure a (e.g.) Haskell program using a suitable development environment e.g. Stack</li> </ul>				
<b>Upon the end of the module the students will have attained the following personal and social competencies...</b>				
<ul style="list-style-type: none"> <li>• to write elegantly constructed haskell functions including use of recursion.</li> <li>• to organise a collection of such functions to build a simple (to intermediate) Haskell program.</li> <li>• to apply Functional Programming Techniques to problems of simple to intermediate complexity</li> <li>• to present a completed a developed program (of simple to intermediate complexity)</li> </ul>				
<b>Textbook/teaching material</b>				
Graham Hutton, Programming in Haskell (2nd Ed) Cambridge University Press. ISBN: 978-1316626221				

Note: this is not the official course descriptor according to the "Studien- und Prüfungsordnung" (SPO)