Computation is a valuable tool that continuously informs and reshapes the field of design. The use of algorithms inherently impacts how designers approach architecture, structure, and sustainability, revealing entirely new forms, functions, and spatial aesthetics. This seminar will introduce the fundamentals of current computational design methods, underlying logic and computational thinking, and applications in architecture and related design fields. The course will begin with an overview of computational logic and a brief refresher of parametric design using the visual programming language Grasshopper. The central chapter of the course will introduce the python programming language, which gives access to methods of customized coding for integrating practical mathematics and machine learning algorithms. Together, these software tools will be used to demonstrate how computation and data science empower geometry generation, optimization, and design exploration. The course is structured around a weekly lecture and exercise that will be reviewed and discussed during a lab session. Students will be evaluated based on their cumulative performance in the exercises and the presentation of a final design project utilizing a combination of the methods learned in the course.

The course lectures and labs will be in person. Students are expected to have a working knowledge of Rhino and Grasshopper and must complete a set of online tutorials before the first class. A laptop computer with Windows 10 (or higher) and Rhino 7 are required. The thoughtful and effective use of AI as a resource for code generation is actively integrated into course assignments.

Computational design is highly recommended as a prerequisite for the new seminar, Computational [robotic] Fabrication, held in the Spring Term!