PREMISE
From algorithmic design to open source and from virtual reality to robotics, many designers are seeking new tools to explore formal and functional properties as sources of ordering systems. This is an introductory course which focuses on computational techniques in architecture as they specifically relate to design processes and procedures. The topics of the course will borrow concepts from computer science, computational geometry and other fields and adapt them to specific design problems such as design development, fabrication, robotics, material simulation, and environmental analysis. A specific area of interest for the course will investigate the relationship between computational methods and fabrication processes. Using the past as a frame of reference for what is to come, we will examine the history of machine epistemology in architecture and draw from a variety of precedents ranging from early 19th century and to the present and investigate how design actualization relies on available tools and knowledge of techniques.

OBJECTIVE
The objective of the course is to help students become better informed users of digital tools and develop the skills necessary for creating or manipulating computational solutions for specific design problems, which include geometry generation and manipulation, analysis of data and design evaluation. As part of the course, students will acquire some hands on experience in parametric modeling and programming as this is the craft that underpins computational design.

METHOD
Class will meet on Mondays and Fridays. Mondays will be given over to guided workshops paired with an assignment due the following Tuesday. These assignments provide students with the opportunity to learn and practice new tools and techniques. The workshop sessions allow for the demonstration of concepts and methods in an interactive setting. Tuesdays will be given over to lecture and discussion. The lectures will frame larger topics within an architectural and historical context while also providing time for further instruction on complex topics.

The mid-term project asks students to analyze a precedent project and develop as set of diagrams with a parametric model. The final project invites students to collaboratively engage in computational thinking through analogue and digital methods and requires a set of compositions (i.e., drawings, paintings, carvings) produced with the robotic arm.

PREREQUISITES
ARCH_2101/2102 for undergraduates / ARCH_6101/6102 or M2 standing for graduates. A basic knowledge of computers and experience with 3D modeling, particularly Rhinoceros, is highly-recommended. No prior programming experience is expected or required.

CREDIT HOURS / WORK EXPECTATIONS
This 3-credit course requires two hours of classroom or direct faculty instruction and approximately six hours of out-of-class student work each week for approximately 15 weeks. NOTE: six hours is an only an estimate. Students of different abilities may take more or less time to complete their course work, but this is considered the average. Out-of-class work may include but is not limited to: lab assignments, extended projects, required readings and videos, and studying for quizzes and exams.

HELP / TA / OFFICE HOURS
Another goal of this course is learning how to become a more independent computer user. This means learning how to teach yourself and fix your own problems when they arise. We will have
many resources for you, but you are expected to help yourself as much as possible. Solving
problems is 90% of programming! Our course TA should be your first point of contact if you
have any Grasshopper, Rhino, or homework questions. Current fourth year students and
second year M2’s may also be able to assist you with your questions. Google is another
resource. You may contact the instructor, but only after you have gone to other sources first.
Due to the large course size, it is important for everyone not to rely upon a single source (such
as the instructor) for help. The instructor and a TA will be present to assist you with any
technical or other questions you might have during the workshop. If you have questions about
the course content or policies, direct them to your instructor. The instructor will not discuss
grading over email. If you would like to discuss a grade, email to arrange an in-person meeting.
This course does not have set office hours. If you need help, we will try to answer you over
email, or you may arrange an appointment via email.

COURSE SOFTWARE
All required software is available on SoA lab computers, but most students will find that
acquiring student versions of the required software will help them to execute their coursework.
Rhinoceros (Rhino) is required for the course. Grasshopper (grasshopper3d.com) is a free plugin
for Rhinoceros that you will also need. Be sure to have it installed on your machine in
preparation for class. Please be prepared to bring your laptop to each class.

COURSE VIDEOS
Course videos will be uploaded to Canvas or a youtube link will be provided. The purpose of
these videos is to give you basic instructions in how to operate Grasshopper in preparation for
the week’s lessons. It is expected that you will watch the videos and practice the necessary
commands and techniques. In this sense, the videos should be considered part of your weekly
assignments. Failure to keep up with the tutorials will hurt your ability to perform well in class,
and will therefore affect your performance for the semester.

ACADEMIC INTEGRITY
All written and graphic submittals, in-class presentations, and other academic tasks should be
your individual and original work unless specifically noted as group projects. No cheating. No
plagiarism. It is assumed that you are aware of and will comply with the spirit and specifics of
the UNC Charlotte Code of Student Academic Integrity, which is available online at:
Integrity will result in disciplinary action. Faculty may ask students to produce identification at
examinations and may require students to demonstrate that graded assignments completed
outside of class are their own work.

DISABILITY
If you have a diagnosed disability which influences your ability to learn or have your work
assessed in the classroom, all efforts will be made to meet your needs. Please provide a copy of
your Letter of Accommodation from the UNC Charlotte Office of Disability Services by the end
of the second week of classes. Their office is located in 230 Fretwell and more information is
available online at: http://www.ds.uncc.edu/. All information about your disability and
accommodations will remain confidential. Please see the instructor if you are interested in being
an official scribe (note taker) for this course. Your notes will be made available to others in the
class with special needs.

ATTENDANCE
In order to fully benefit from and participate in this course, attendance is required. Two (2)
unexcused absences automatically lower your final grade one letter grade. More than two (2)
unexcused absences will constitute grounds for automatic failure of this course. Documentation
of excused absences must be submitted in writing and show evidence of the medical or family
emergency. When possible, notify your instructor as early as possible in advance of a potential
absence.
LATE WORK
Late work will not be accepted, and will not receive credit. A printed hard copy of each submittal is due at the beginning of the class period on the due date indicated in the class schedule. If you are unable to complete an assignment due to an excused absence, notify the professor on the due date and turn in the assignment at the next class meeting. Failure to turn in two assignments on their due dates is grounds for automatic failure of the course.

GRADING
Final course grades are based upon the following rubric:
- 10% Attendance and Participation
- 40% Weekly Assignments
- 25% Midterm Project
- 25% Final Project

All courses in the SoA are governed by the rules and regulations of UNC Charlotte as stated in the University Undergraduate and Graduate Catalogs. For more information about these polices, please refer to the appropriate catalog, which can be found online at: http://www.provost.uncc.edu/catalogs/2007%2D2009/ (undergrad) and http://www.uncc.edu/gradmiss/gs_catalog.html (grad).

Grading of courses conform to the following grading scales and values:

<table>
<thead>
<tr>
<th>Undergraduate Scale &amp; Values</th>
<th>Graduate Scale &amp; Values</th>
</tr>
</thead>
<tbody>
<tr>
<td>A 90-100</td>
<td>A 90-100</td>
</tr>
<tr>
<td>Good 80-89</td>
<td>B 80-89</td>
</tr>
<tr>
<td>Fair 70-79</td>
<td>C 70-79</td>
</tr>
<tr>
<td>Passing 60-69</td>
<td>U 69 &amp; Below</td>
</tr>
<tr>
<td>Failing 59 &amp; Below</td>
<td>Unsatisfactory</td>
</tr>
</tbody>
</table>

NAAB ACCREDITATION
The School of Architecture at UNC Charlotte maintains accredited status through the National Architectural Accrediting Board (NAAB), which reviews the curriculum, facility, faculty, and program resources annually, and conducts an intensive site visit. As part of this review, the NAAB will review student work produced in courses throughout the curriculum. This course will include content related to the following NAAB Student Performance Criteria:

A.1 - Professional Communication Skills - Ability to write and speak effectively and use representational media appropriate for both within the profession and with the general public.

A.4 - Architectural Design Skills - Ability to effectively use basic formal, organizational and environmental principles and the capacity of each to inform two- and three-dimensional design.

A.5 - Ordering Systems - Ability to apply the fundamentals of both natural and formal ordering systems and the capacity of each to inform two- and three-dimensional design.

A.6 - Use of Precedents - Ability to examine and comprehend the fundamental principles present in relevant precedents and to make informed choices about the incorporation of such principles into architecture and urban design projects.
SCHEDULE

Week 1 // F 8.26 // Lecture: Introduction to Computational Methods Thinking

Week 2 // M 8.29 // Assignment 1 Workshop
    // F 9.02 // Lecture: Machine Epistemology in Architecture

Week 3 // M 9.5 // Assignment 2 Workshop
    // F 9.9 // Lecture: Digital Craft and Ruskinianism

Week 4 // M 9.12 // Assignment 3 Workshop
    // F 9.17 // Lecture: History of Methods

Week 5 // M 9.19 // Assignment 4 Workshop
    // F 9.23 // Lecture: Geometry & Encapsulated Knowledge

Week 6 // M 9.26 // Assignment 5 Workshop
    // F 9.30 // Lecture

Week 7 // M 10.03 // Precedent Research and Crit
    // F 10.07 // Lecture

Week 8 // M 10.10 // Mid-Term Project: Precedent Research and Parametric Model
    // F 10.14 // Lecture

Week 9 // M 10.17 // Assignment 6 Workshop
    // F 10.21 // Lecture: Intro to Scripting in C#

Week 10 // M 10.24 // Assignment 7 Workshop
    // F 10.28 // Lecture: Loops and Conditional Statements

Week 11 // M 10.31 // Assignment 8 Workshop
    // F 11.04 // Lecture: Material Agency

Week 12 // M 11.07 // Final Project: Analogue Studies
    // F 11.11 // Lecture: Intro to Robotics

    // F 11.18 // Lecture: Robotics and Advanced Manufacturing

Week 14 // M 11.21 // Final Project: Drawing Taxonomy
    // F 11.25 // Thanksgiving Break

Week 15 // 11.28 // Final Project: Robotic Drawings

The schedule is subject to adjustment during the course of the semester based on the progress of the entire group. Any schedule conflicts due to religious or health reasons, etc. should be brought to the attention of the course instructor during the first week of the semester.