MEEM 5990 Design Automation: Theory and Implementation
Course Notes

I. Introduction

I.1. Welcome.

Design Automation is a field of study related to development of technology to automate the design of products from concept design, through layout design, and detail design. We are specifically interested in Design Automation for the design of mechanical products. Areas of study in Design Automation include topics such as Geometric Modeling, Virtual Reality, Engineering Knowledge representation and usage, Collaborative Design, Design for Manufacturing, and Optimization. The people involved in Design Automation come mainly from mechanical, aerospace, and civil engineering, architecture and computer science. They come mainly from universities, manufacturing companies or CAD-related software companies.

For us, "Design Automation" will mean the same thing as "Computer-Aided Design," however this course is different from MEEM 4403 Computer-Aided Design Methods in that this course focuses on the **development** of CAD software (and theories) as opposed to the **use** of CAD software. This course should also not be confused with Manufacturing Automation, whose output is a physical manufactured product rather than a design. This course will not cover all of the areas of Design Automation. Instead it will to be a practical course you can use to help you in your research, to create models and implement analyses on the computer, and to possibly help you in the future, should you wish to automate an intellectual process. The course will involve a lot of programming, but will also make use of a lot of existing software programs and libraries. In particular, we will be writing programs that interface with Unigraphics.

The course does not require you to have any programming experience, and the first thing we will cover is how to program in C++. Then we will cover data structures and algorithms used in CAD software. Of primary consideration will be data structures and algorithms used in geometric modeling. We will then examine topics such as the internet, optimization, engineering knowledge representation, feature recognition, and automated design synthesis.

Benefits from the course may come in two ways. First, the information learned in the course may help you in your research. If your research involves some kind of physical modeling or simulation, you generally need to describe the geometry of the problem first, in order to create the model. This can be done using software such as I-DEAS or Unigraphics. You could then write software to read this geometry and use it to do your analysis. Similarly, if your research is in the area of design, you will likely need to evaluate a part or assembly. This could also be done by taking the information from I-DEAS or Unigraphics and processing it as required.

Secondly, what you learn can be applied in industry. A surprisingly large number of companies have employees writing CAD-related programs. These companies fall into two categories: software companies and manufacturing companies. Some software companies such as ThermoAnalytics in Hancock, Michigan develop software to model some kind of physical affect. ThermoAnalytics markets customized thermal analysis software, which calculates the heat transmitted to objects by radiation. This software is used by automobile manufacturers to predict air conditioning loads and by the military to analyze heat signatures from vehicles. A significant part of the programming effort is taken up in using geometric information to create the numerical model. Another company Emergent Systems of Dearborn, Michigan develops engineering knowledge base software for automobile manufacturers to track knowledge (rules) for how cars should be designed and built. Most manufacturing companies also develop software for internal use, to automate processes that had previously been done manually. For example, at Ford, software has been written to automatically evaluate door gap geometry and door seals from I-DEAS files to determine whether excessive force will be required to close the door, or whether wind noise will penetrate through the seals. Companies such as Ford also develop software to automatically generate manufacturing process plans. At Boeing, software has been written to automatically find the optimal skin thicknesses and stiffener dimensions for fuselage sections. Once this has been done, other software automatically creates the solid models, assembly drawings, and detailed manufacturing drawings. A process that used to take months, is now done by a computer that runs overnight.

I.2. History of CAD

The last few decades have seen great increases in technology, which has lead to the introduction of many new and improved products. As well, these products are now coming to market faster and faster. Because of the ease of product development we are seeing increased specialization of products, to meet customer needs and desires more specifically. These tremendous capabilities are likely to speed up our ability to pursue amazing activities such as space exploration.

These great improvements in design capabilities have been made possible by only a small number of major scientific break-throughs and a large number of minor advancements.

Alan Turing (1912-1954) was a British mathematician who, in 1936 developed a conceptual framework for algorithmic computation called the Turing machine. The purpose of the machine was to decipher coded German transmissions during the war. The basic concept was that the machine could read instructions from a tape (the program), process the instructions and output results. This was to avoid the drudgery and mistakes that were made from doing these calculations by hand. Though his machine was mechanical, it had all the fundamental elements of the modern computer: memory, a program, and a processor.

ENIAC, the Electronic Numerical Integrator and Computer, was developed in 1946 at the University of Pennsylvania. This was the first electronic computer. It used 17468 vacuum tubes for transistors. Innovations were that it integrated diverse technical components and design ideas. It was also very reliable compared to existing computing machines, and fast for its time.

Ivan Sutherland, a Ph.D. student at the Massachusetts Institute of Technology, wrote a Ph.D. dissertation in 1963 titled "Sketchpad: A Man-Machine Graphical Communication System". His idea was to use a graphical interface to directly enter engineering drawings into a computer. This was a radical idea because most computers at the time were only batch-processing systems (using punch cards). The system that Sutherland developed consisted of a nine-inch display, a light-pen and a bank of switches to control the input modes. Sketchpad was, literally, the first Graphical User Interface (GUI). It pioneered many idea that we take for granted today, such as "snap-to" guides to ensure perfectly straight lines, geometric constraint solving, zooming in and out, rubber-banding so you can see lines move when you drag them, and using data structures in memory to store attributes of objects. He later founded Evans & Sutherland in 1968, and pioneered the first head-mounted virtual reality display. In 2002 he was a vice president and senior research fellow at Sun Micro-Systems.

The development of Solid Modeling data structures was also a remarkable stride in CAD's history. In the early 1960's, two camps of researchers existed: those that believed that solids should be described by their boundaries and those that believed solids should be described as a combination of primitive Euclidean objects. In 1973, Ian Braid developed one of the earliest boundary-representation solid modelers, BUILD, as part of his Ph.D. research at the University of Cambridge Computer Lab. This later lead to the development of one of the first commercial modelers ROMULUS. In the United States, at the University of Rochester, Herbert Voelcker, Aristides Requicha, and others developed the PADL-1 system from 1975-1977 and the PADL-2 system was released for commercial use in 1981. The PADL systems combined Constructive Solid Geometry and Boundary Representations in a hybrid system. Many current CAD software systems use ACIS or Parasolid solid modeling cores, which have roots in these earlier systems.

I.3. Assignment 1: Hello world (console)

Type in, compile and run the following C++ program to print "Hello world!!" in the Unix terminal window.

```
#include <iostream>

using namespace std;

int main()
{
  cout << "Hello world!!" << endl;
  return 0;
}
```