CAD – The Creative Side
An educational experiment that aims at changing students' attitude

Maria Gabriela Caffarena Celani, PhD
Department of Architecture, MIT, Cambridge, MA, USA.
e-mail: celani@alum.mit.edu

Abstract

The present paper describes an innovative design education system tried out at two different architecture schools in Brazil, with opposite approaches to the use of CAD. The experimental courses had two main goals: (1) to explore the use of logical operations in design, such as symmetry, recursion, parameterization, and combinatorial analysis, and (2) to apply these techniques with the use of the computers, using CAD not only as a representational tool, but rather as an explorative, customizable and programmable design aide for the creative process. The experiments resulted in a number of interesting compositions, design projects and programs, and assessment questionnaires revealed a real change in students' attitude towards the use of CAD in architecture. The experiments related were the field research part of a Ph.D. thesis defended at MIT in July 2002.

The present work had the support of CNPq, a Brazilian entity devoted to the scientific and technological development.

1. Introduction

In the last forty years of CAD development, the original purposes of Computer Aided Design have almost been lost. Nowadays its use in architecture is practically restricted to drafting and representation, with little emphasis on the design process. Mitchell (1990) identifies an inflection point in CAD history with its popularization in the 80's, targeting the emergent PC market, and suggests that at after that point "the wider possibilities were largely ignored"(p. 483). In fact, with the standardization of CAD packages, they were turned into drafting versions of word-processing software. As a consequence we end up using an extremely powerful machine below its capacities, and keep doing repetitive tasks while we could make better use of our time. Most of all, we waste the possibilities of finding new shapes with the help of complex computations, which are impossible to envision when working by hand.

The present paper describes two experimental instances of an innovative CAD education system developed as a Ph.D. thesis for the Design & Computation program at MIT's School of Architecture and Planning. The system's approach is similar to Mitchell, Ligget and Kvan's in *The art of computer graphics programming* (1987) and is inspired by Stiny's theory of shape grammars (1975). It is based on six topics from computational design, put into practice through a series of exercises that include using a CAD's standard commands and programming to create new design environments. The six topics - symmetry, recursion, parametric shapes, shape generation, algorithmization of design processes and emergent shapes - consist of overlapping, inter-related concepts that reinforce the understanding of the computational aspects of design. For example, symmetry can be generated through the recursive application of a design rule, while the generation of design alternatives can be obtained after the parameterization of a shape and the substitution of its variables by ranges of values within the problem's constraints.

2. Context

The present experiment has been conducted in two top-ranking architecture schools in Brazil: the School of Architecture and Planning at the University of São Paulo (FAU-USP), which became independent from the engineering school in the late 60's, and the recently founded architecture course of the School of Civil Engineering at the University of São Paulo at Campinas (FEC-UNICAMP). While the former has shown a peculiar resistance to the inclusion of computer-based tools for architecture, the later has incorporated IT subjects in its curriculum since its very beginning. While FEC-UNICAMP has mandatory CAD courses from the very first semester, at FAU-USP the only two subjects offered are elective courses with limited enrollment, usually attended by students from upper grades. Despite that difference, in both schools the IT subjects focus on the use of the computer as a representational tool.

The CAD software used in both cases was AutoCAD 2000 ADT, which includes an interactive VBA development environment. The group of students attending the experimental course at FAU consisted of 19 undergraduate architecture students, all of them with some prior experience in AutoCAD, but none with prior experience in VBA programming. At UNICAMP, the group attending the experimental course consisted of 7 undergraduate architecture students, 6 undergraduate civil engineering students, 4 architecture graduate students and 2 professional architects, all of whom with some prior experience in AutoCAD, but also none with prior experience in VBA, although the undergraduate engineering students had some prior experience with other programming languages.

The course at FAU-USP happened as a series of 6 weekly sessions during a school semester , while at UNICAMP it was taught as a Summer course with daily meetings for two weeks. In both cases, the course materials were grouped under web sites, which included course readings, directions to exercises, technical instructions, lectures' slides and downloadable programs (see http://web.mit.edu/celani/www/fau/index.htm and http://web.mit.edu/celani/www/unicamp/index.htm). Besides, as the courses evolved, all the exercises presented by students were gradatively incorporated to the web site for classroom review. The two courses happened during the first semester of 2002.

3. Course description

In both cases, each session started with a lecture in which concepts were presented and buildings from the architectural repertoire were analyzed in terms of each concept. The second part of the sessions was always used for classroom exercises. Students were also given a number of reading materials related to each topic, which offered a theoretical support to the course.

For the first topic (Symmetry), students used AutoCAD's commands in a innovative way. With the use of multiple, dynamically updated views of the design space, students were able to design symmetric patterns in an environment that emphasized the whole instead of the units. During the study of topic 2 (Recursion), exercises were based on the use of Recursion Assistant (Figure 1), a VBA application especially developed for the courses. In topics 3 through 5 (Parameters, Generation and search and Algorithmization), small programs that incorporated each theoretical concept were first presented and used by students, then thoroughly explained, and finally modified by them. VBA programming was introduced in these 3 sessions, starting with the explanation of simple, pedagogical, sample programs, and the concepts behind them. In topic 6 (Emergent shapes) students started by generating 2D and 3D rule-based compositions with Recursion Assistant and then used standard CAD commands to perform Boolean operations on them, unveiling novel shapes.

Due to time constraints, only at FEC-UNICAMP students were asked to develop a final project, most of which consisted of VBA programs for automating design procedures or modeling parameterized versions of existing buildings. At FAU-USP, on the other hand, students were asked to develop small architectural design exercises as homework after each weekly session.

4. Results

The results of the two experimental courses was an impressive number of beautiful compositions and thoughtful programs. The exercises presented could be separated in five categories:

- Creative abstract compositions based on symmetry, recursion, parameterization and emergence (Figures 2 through 5).

- Small programs developed as tutorial exercises. Although these were in general not particularly creative, some students were able to propose interesting variations of the programs suggested, such as those in Figure 6.

- Images of architectural examples of symmetry, recursion and parametric shapes brought to class by students as a research homework, which provided a clue to how well students were assimilating the concepts introduced in each session.

- Architectural design exercises based on symmetry, recursion, parameterization and emergence, only at FAU-USP (Figure 7).

- Free-style final projects, only at FEC-UNICAMP (Figure 8).

Most of the final projects presented at FEC-UNICAMP consisted of VBA applications for different purposes, such as an automatic layout generator (based on a program provided in the course), an alternative generator for land-use, a stair-case 3D model generator, and programs for generating parametric versions of buildings by Brazilian architect Oscar Niemeyer. One of the best final projects was presented by an M.Sc. student with a background in architecture, who compared two designs by American architect Frank Lloyd Wright (a church and a synagogue) and developed a program to model both using the same algorithm, but taking different parameters (Figure 8). This student was able to analyze the two buildings from a computational point of view, understand and relate their geometric construction processes, and generate a common, parameterized description of both.

5. Discussion

The careful analysis of the exercises presented allowed to demonstrate students' understanding of the design environments and the concepts introduced in the educational experiments described above. Those exercises allowed students to put into practice the computational design theories learned, using techniques and resources at their reach. The two case studies proved also that the proposed system is flexible enough to be applied in different formats and for different background students, even when they are mixed in the same group, which was the case at UNICAMP.

Assessment questionnaires showed how the courses were successful in changing students' attitude towards CAD. Most students reported how they used to see CAD simply as a representation tool and how they started seeing it as a real design aide after the course. Although most students were not able to immediately apply what they learned in the experimental courses in studio subjects – which was due to different constraints, such as the unavailability of computers in the classrooms – they reported a strong conceptual influence to their design work. This shows how a shift in the way CAD is taught can also serve as an opportunity to introduce computational concepts in design education.

A comparison between the level of achievement of students with different technical skills was possible because of the different emphasis given to CAD at UNICAMP and FAU-USP. Apparently, the less familiar students were with CAD software, the harder it was for them to start seeing new possibilities in the use of computers in design. FAU-USP students, in general less skilled in CAD, often complained when they were asked to develop small programs in VBA, while UNICAMP students were more open to learning the new technique. For that reason a more philosophical approach was tried at FAU-USP and students were asked to develop architectural design projects using the concepts learned, instead of focusing too much in the programs. At UNICAMP, on the other hand, developing an architectural design project was one of the suggested themes for the final project, and yet none of the students – either architects or engineers – chose that way. All of them wanted to develop their own programs (except for an engineer who made an interesting computational interpretation of the work of Italian/Brazilian painter Alfredo Volpi).

A few unexpected observations could also be made during the two experimental courses, such as the efficacy of the two different formats. The summer course's compact format was probably more effective in terms of the assimilation of the technical contents, especially the programming language. In the more extensive format tried at FAU-USP, students seemed to forget the technical details learned from one week to the other, although they probably had more time to reflect on what was being presented at the conceptual level.

Although the results of the experiments described above cannot be over-generalized, they provide a preliminary conclusion of what kind of response can be expected from students in similar contexts. A deeper understanding of how CAD-implemented computational design education may influence the quality of designs produced by students is the next step in the research, but one that will demand a longer observation time and a close collaboration of studio instructors.

6. Acknowledgements

I would like to thank Professor Doris Kowaltowski from UNICAMP and Prof. Marcelo Giacaglia, from FAU-USP, for providing me with the opportunity to test my experimental course with their students. I would also like to thank all of the students listed below for participating in the experimental courses and contributing to my research with their talent and enthusiasm. Finally, I would like to thank CNPq, the Brazilian National Council for Scientific and Technologic Development, for funding this research.

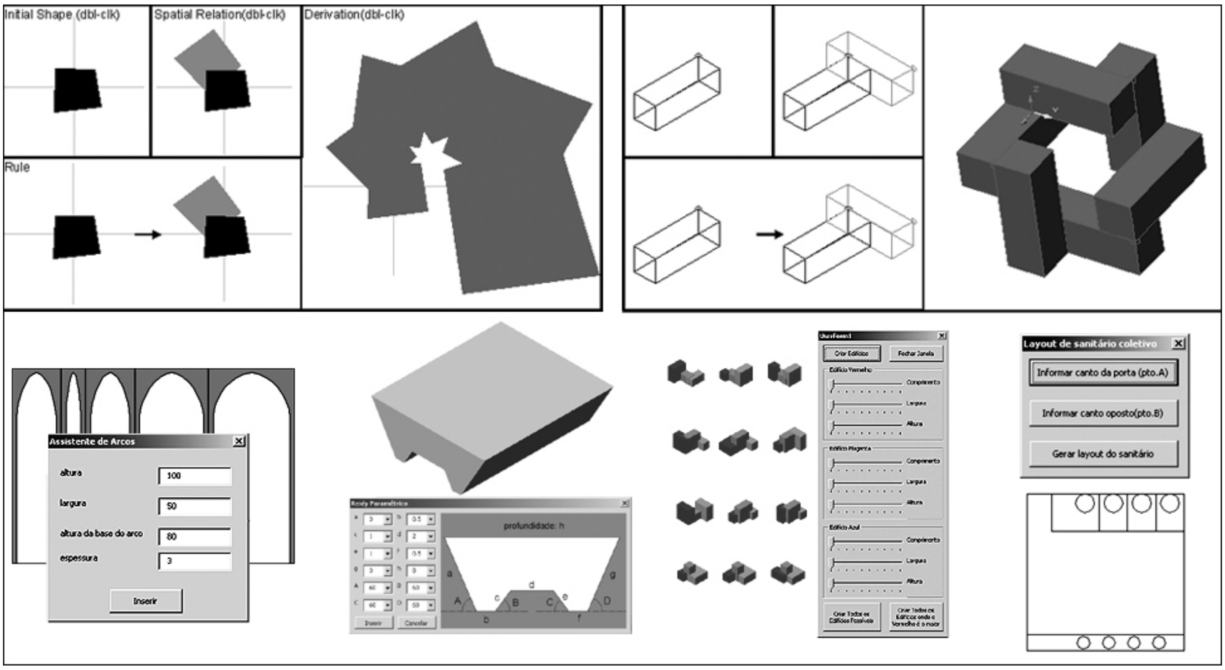
7. List of students*

UNICAMP: Amanda Pietro Petter, Ana Lúcia Harris, Ana Maria Monteiro, Christian Dittz, Daniel da Rocha, Daniel Moreira, Fernando Basilio, Fernando Ribeiro, Gabrielle Damaso, Giovana Bianchi, Lauro Luiz Fco Fo, Livia Carvalho Berriel, Marina Otaviano, Patrícia Dias Falcão, Paula Roberta Baratella, Roberto Itapura, Sérgio Luiz Montagner, Wanessa Watrin.

FAU-USP: Ana Carolina Salomao, Chen Chih Cheng, Daniela da Costa, Daniela Vaz, Fabio Augusto Bellini, Fernando Vargas, Juliana Beividas, Laila de Andrade, Leandro Robles, Marcelo Nakazaki, Marcos Machado, Min Kyung Lee, Renata Figueiredo, Rodrigo de Azevedo, Stella Tomiyoshi, Thiago Lessa.

* Only the students who authorized the publishing of their names are cited here.

8. Figures



Initial Shape (dbl-clk)
Spatial Relation(dbl-clk)
Derivation(dbl-clk)
Rule
profundidade: h
a
b
c
d
e
f
g
A
B
C
D
Inserir
Cancelar
UserForm1
Criar Edifícios
Fechar Janela
Edifício Vermelho
Comprimento
Largura
Altura
Edifício Magenta
Edifício Azul
Criar Todos os Edifícios Possíveis
Criar Todos os Edifícios onde o Vermelho é o maior
Layout de sanitário coletivo
Informar canto da porta (pto.A)
Informar canto oposto(pto.B)
Gerar layout do sanitário

Assistente de Arcos

altura 100

largura 50

altura da base do arco 80

espessura 3

Inserir

Reidy Paramétrico

Figure 1: **VBA applications** especially developed for the experimental courses: 2D and 3D Recursion Assistant, Parametric Arches Assistant, Modernist Profile Assistant, Alternative Generator and Bathroom Layout Generator.

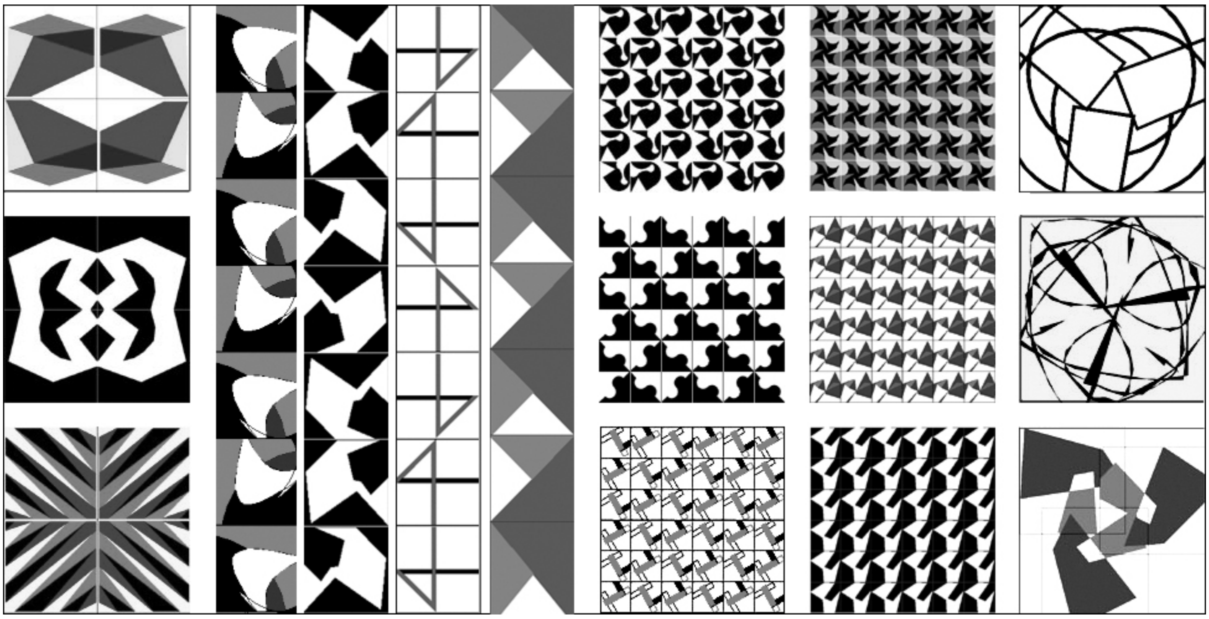


Figure 2: **Symmetry** exercises: **bilateral** (from top to bottom, left to right: Marcelo Nakazaki, Leandro Robles, Marcos Machado), **frieze** (Renata Figueiredo, Daniela Costa, Min Lee, Marcelo Nakazaki), **wall-paper** (Flávio Tanabe, Daniela Vaz, Renata Figueiredo, Daniel Rocha, Gabrielle Damaso, Marina Otaviano) and **cyclic** (Fernando Mello, Chen Cheng, Marcos Machado).

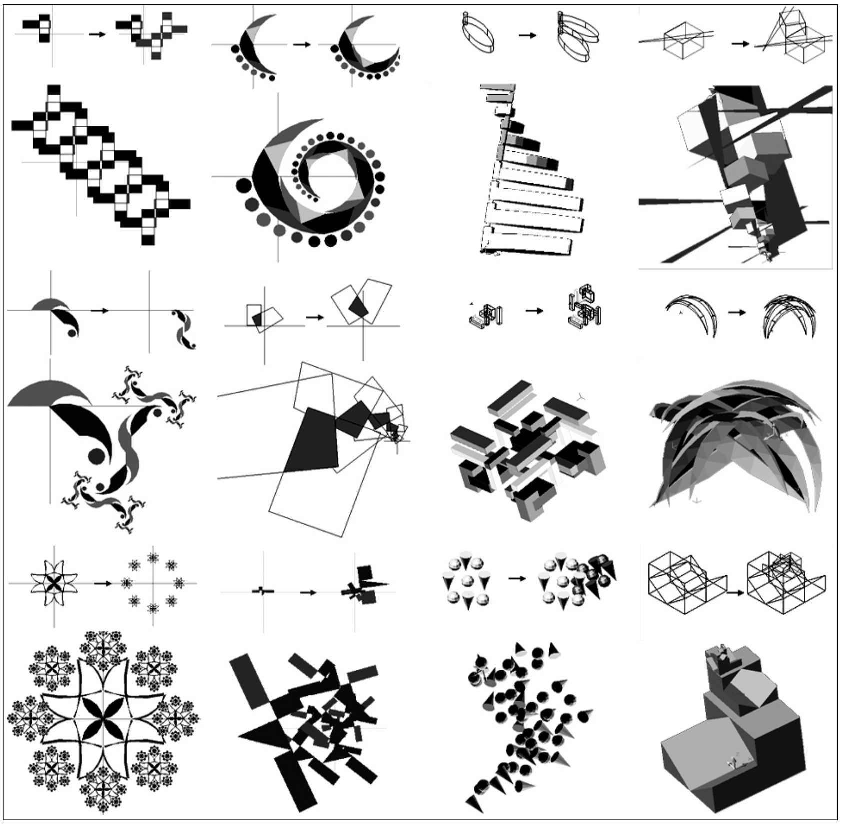


Figure 3: **Recursion** exercises: **2D** (from top to bottom, left to right: Renata Figueiredo, Núbia Bernardi, Gabrielle Damaso, Giovana Bianchi, Daniel Rocha, Fábio Bellini) and **3D** (Fernando Basilio, Marina Otaviano, Fernando Ribeiro, Ana Goes, Daniel Moreira, Gabrielle Damaso).

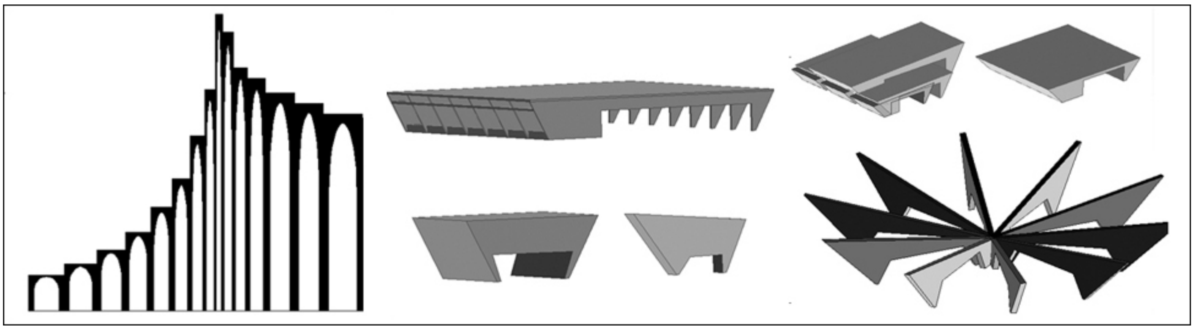


Figure 4: **Parameterization** exercises (Fábio Bellini, Marina Otaviano , Paula Baratella,).

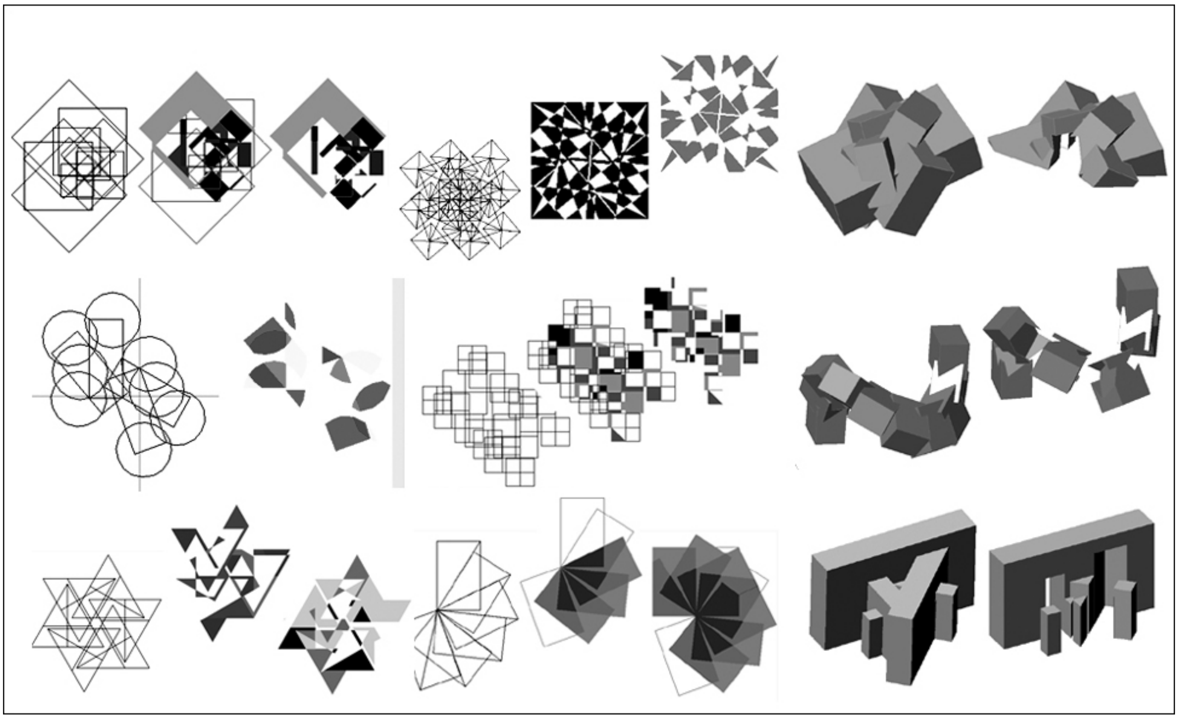


Figure 5: **Emergence** exercises: **2D** (top to bottom, left to right: Fernando Basilio Renata Figueiredo, Fernando Basilio, Gabrielle Damaso, Alessandra Arenales, Daniel Moreira) and **3D** (Stella Tomiyoshi, Fábio Bellini, Leandro Robles).

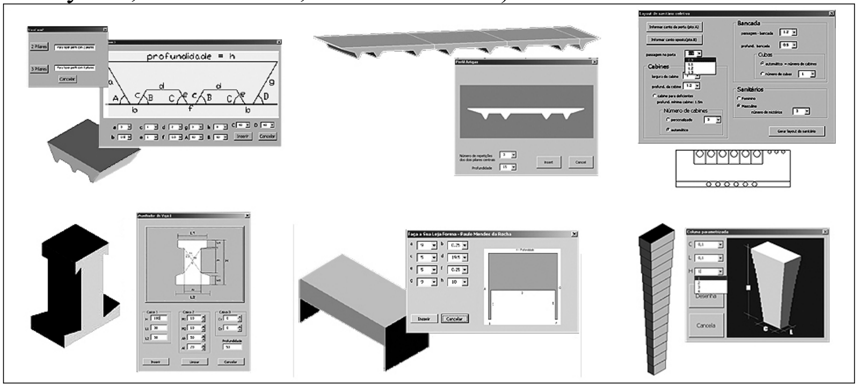


Figure 6: Some parameterization programs developed by students (top line, left to right – programs evolved from classroom exercises: Patrícia Falcão, Marina Otaviano, Marina Otaviano; bottom line, left to right – original programs by students: Min Lee, Marcelo Nakazaki, Wanessa Watrin).

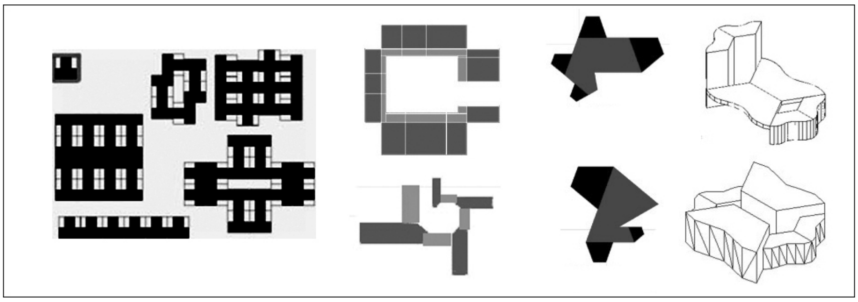
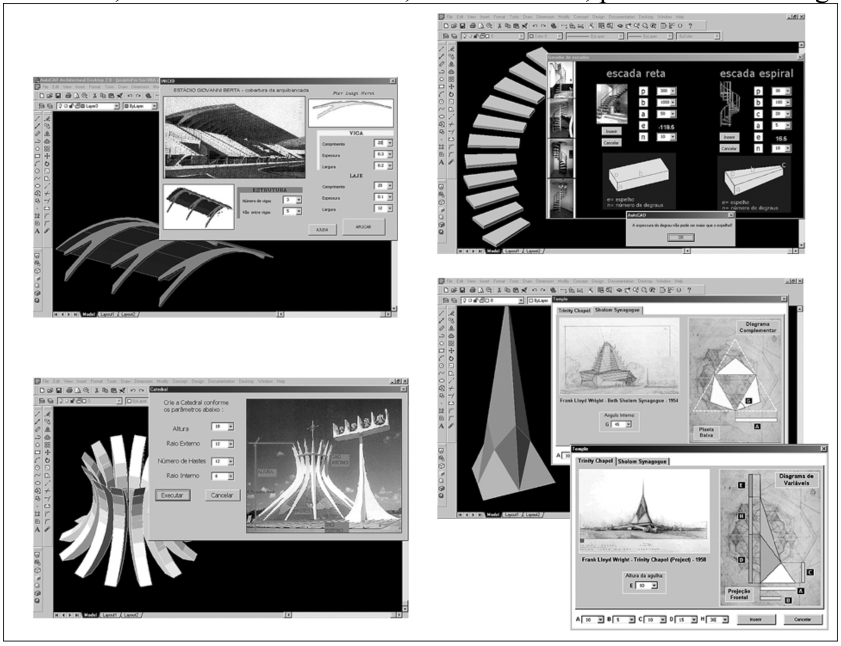


Figure 7: Architectural design exercises presented by FAU-USP students (top to bottom, left to right: symmetric housing, Daniela Costa; parametric school, Stella Tomiyoshi; recursive museum, Marcelo Nakazaki; recursive movie theater, Daniela Costa; parametric buildings, Daniela Costa).



INICIO

ESTÁDIO GIOVANNI BERTA - cobertura da arquibancada

Pier Luigi Nervi

VIGA

Comprimento 30

Espessura 0.3

Largura 0.2

LAJE

Comprimento 20

Espessura 0.1

Largura 12

ESTRUTURA

Número de vigas 3

Vão entre vigas 5

Gerador de escadas

escada reta

p 300

b 1000

a 50

e -118.5

n 10

escada espiral

p 30

b 100

c 20

a 5

e 16.5

n 10

e= espelho
n= número de degraus

AutoCAD

A espessura do degrau não pode ser maior que o espelho!

Catedral

Crie a Catedral conforme os parâmetros abaixo :

Altura 10

Raio Externo 12

Número de Hastes 12

Raio Interno 6

Templo

Frank Lloyd Wright - Beth Sholom Synagogue - 1954

Angulo Interno:

G 45

Frank Lloyd Wright - Trinity Chapel (Project) - 1958

Altura da agulha:

E 10

A 10 B 5 C 10 D 15 H 30

Figure 8: Some examples of final projects at FEC-UNICAMP (top to bottom, left to right: program for modeling parametric variations of a structure by Nervi, Fernando Basilio; program for modeling parametric variations of a Brasilia's cathedral, Patrícia Falcão; program for modeling stairs, Paula Baratella and Giovanna Bianchi; program for modeling parametric variations of a church and a synagogue by Wright, Daniel Moreira).

9. References

Mitchell, W. J. (1990) "A new agenda for computer aided-design". In The electronic design studio. W. J. Mitchell & M. McCullough (Eds.) 1-30. Cambridge, MA: The MIT Press.

Mitchell, W. J., Ligget, R. S. & Kvan, T. (1987) The art of computer graphics programming. New York: Van Nostrand Reinhold.

Stiny, G. (1975) Pictorial and formal aspects of shapes and shape grammars - on computer generation of aesthetic objects. Basel: Birkhäuser.