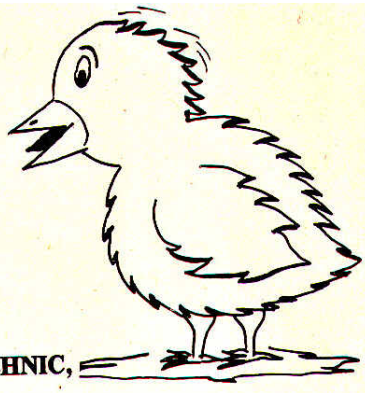
Architectural software on the cheap



PAUL COATES,
LECTURER AT DEPARTMENT OF ARCHITECTURE, LIVERPOOL POLYTECHNIC,
53, VICTORIA STREET, LIVERPOOL L1 6EY.

LIVERPOOL Polytechnic Department of Architecture ordered its first micro (a 7k PET) in February 1978. This arrived in July of that year. Since then the work of the Department has developed and we now have two 32k PETs, Commodore disk and printer and a Hewlett Packard Plotter. We are now approaching the point where we can offer a reasonable service to our students and staff, with two part-time research assistants and *2+* , years of work, and we are planniog to develop a range of educational software that will be useful to all of the 20 or so schools of architecture with PETs or other similar machines.

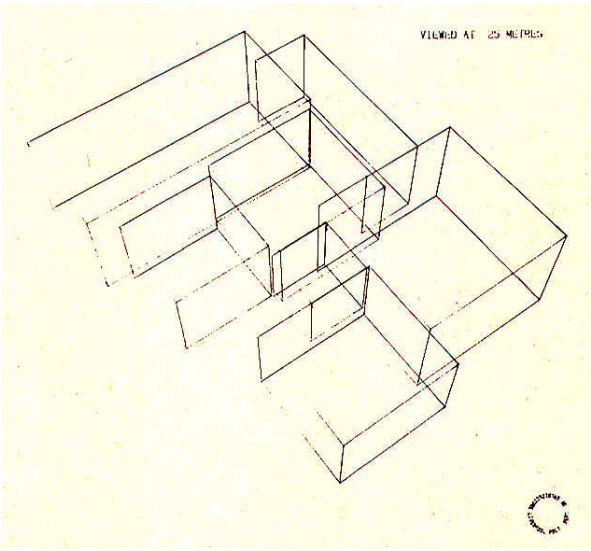
I had decided to write this article by putting together all the notes and explanations that I have written over the last two years about our aims and intentions, usually for the benefit of some worthy grant awarding body or committee. However, it would be a pity to add to the enormous literature of what people would like other people to think they're doing, and instead I will simply talk about those programs that we have working now.

These programs fall into two main categories. Those that are supposed to get the student to think ('teaching programs') and those that do his thinking for him (programs). The former, which I find most interesting, are harder to well to students, who not surprisingly resent being made to think by a machine. I try to point out that they don't mind being forced to think by a piece of cardboard when they are playing Monopoly, but of course it takes luck and inspiration to be instructional and amusing at the same time.

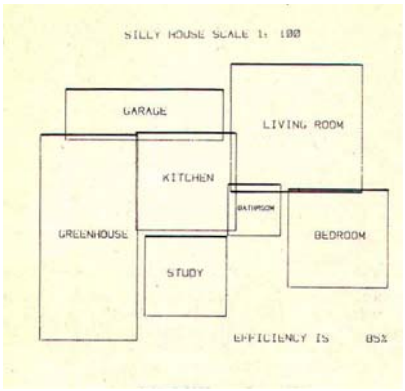
Students are reluctant to invest time and energy in using the computer unless they can get something very tangible out of it, and so I will draw a veil over the PET version of the Open University Problem Identification Game, or our general purpose Multiguess (based on the animal game) pausing only to mention our version of PEGASUS (written for PDP 11 by Mildred Shaw of Middlesex Polytechnic).

Pegasus was originally written as a psychological testing procedure but it can be used with students to explore the way they. think about a problem. At the end of a half hour session they come away with a neat pair of cluster diagrams showing the results of a statistical analysis of their responses. The only other half decent teaching programs we have are Fawlty Towers and Planner. Fawlty Towers is actually quite a good game, but the intention was to use it as a way of explaining aspects of graph theory, which would be used in Planner. The hunt the dead body business (for it is on the surface a sort of Cluedo) is based on a random graph - a set of rooms which may or may not interconnect. Once the body is found there is a quiz about the plan which usually means that, unless you made some kind of map you get poor results.

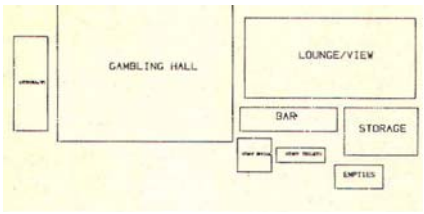
Once you've got the idea across that a plan can be represented as a network of points and lines, which in turn can be represented in a matrix, then it's on to Planner. Here you define the ideal relationships between rooms on a matrix, then assemble a plan on the screen and test the efficiency (ie. how far it corresponds to the 'ideal' relationships). This is all very easy to do (it had better be or else most of the students would walk away) using the joystick (kindly provided with software by John Stout Q.V.) and four keys. There is a menu of commands that allow you to Swap, Alter (shape), Change (position), change the Matrix, draw a perspective (Fig.1)



(of which more later) or file data on disk for use after tea, or plot out the plan with labels of room names and the efficiency (Fig.2).



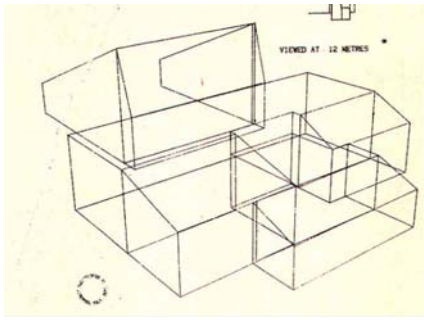
SOTHPORT **PIER SCALE I, 500**



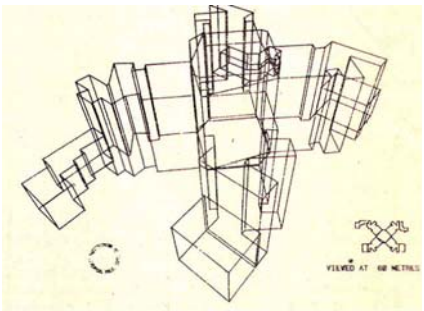
The object of the exercise is to teach students about the use (and abuse), usefulness and limitations of the connectivity matrix. This is the table that defines the 'ideal' relationships and the program allows you to set it up, try out and plan, see the result, move things around, try it again, and change an item in the matrix, try it again and so on.

Most students find it fairly amusing and quickly grasp the circularity of the argument. This years First Year did a traffic survey of the Third floor in our building and from that we input the movement data to the matrix (this was all done with one student at tho PET and the rest looking at the display on a 36" telly .(interface from STACK). We then 'drew' the plan of the Third floor on the screen and tested the efficiency. It came out as a dismal 36%, so they all tried moving things about. we got it up to 60 odd by putting the bogs in the middle of the staff room and moving the stairs next to them. In one afternoon I was able to demonstrate graphically most of the fundamentals of space planning vis a vis circulation, we had lots of chat about the problem and everyone had had a jolly time.

Apart from these game based programs we have a range of programs that cover the wide range of specialist subjects that architecture is connected with, Building Economics. Structures, Light and Heat. and graphical presentation. These are used usually by the specialist staff involved, except for the perspective program. This has been recently written by Anne Scott (our RIBA.funded research assistant) and provides a very easy way of digitising information from a plan of the building, using the plotter, and drawing perspective views (without hidden line removal). The program makes an assumption that building can be defined as
a series of vertical planes with defined top and bottom
edges (which can be sloping) plus a 'roof' of freely defined points if necessary. This allows quite complex buildings to be digitised quite quickly using a user defined set of 10 keys to indicate 'wall types' (i.e, pre. defined top and bottom heights). A series of pitched elements disposed down a slope (as in the drawing. Fig.3) was digitised in a few minutes.

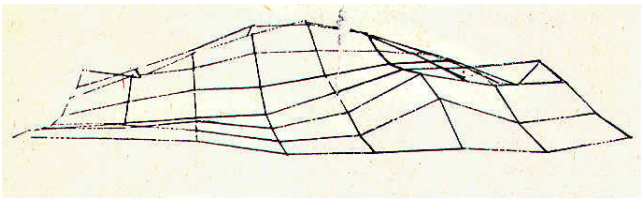


This program is without hidden line removal, but for a student with a fairly complex plan it does offer a
chance to generate 20 or so views in the course of say an afternoon. (Fig.4)



The perspective routines are written as a self contain-ed lump and can be used to provide 3-D views of a variety of objects, hence their appearance in 'Planner'.

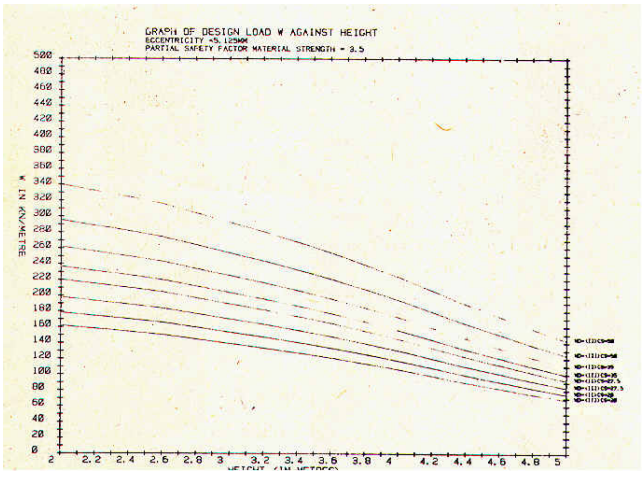
We have recently (Friday 18th) finished a 'site perspec tive' program using the same routines to generate an approximation to a complex curved surface (see fig.6)



using data digitised from a map on the plotter, (the view is of a part ofthe environs of Craigeiver Castle in the Cairngorms - with the vertical scale exagerated three times).

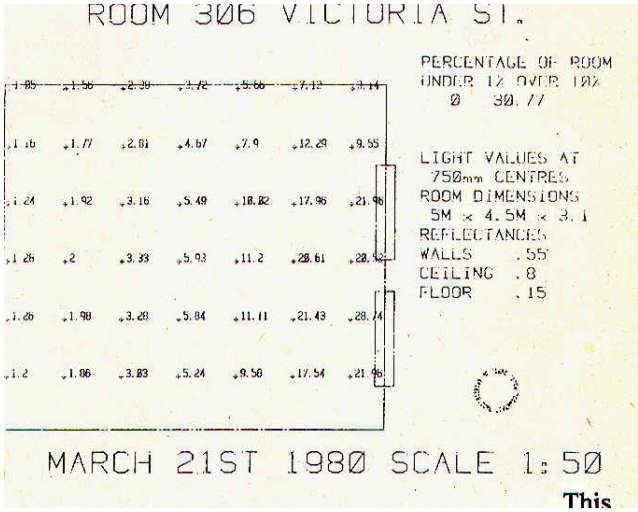
In the last six months Susan Davies (our Nuffield sponsored research assistant) and the staff responsible for teaching building economics have been working on teaching programs to demonstrate some of the subtleties of the inter-relationships between cost factors in building - geometry, glazed areas and cost of walling material for instance. We now have two programs 'Costglaze' and 'Costgeom' which can be used to demonstrate simple relationships in these areas.',

In the general area of structural design, we have at the moment only one rather primitive teaching, programme - Beam Tutor - which graphically shows the bending stresses in a simply supported beam, ,and where the supports can be moved to show the effect; on the stress distribution. There are now two non teaching programs as well, one for steel beam design and one for calculated brickwork. The beam design program useis the disk for holding the tables of section geometry data will determine the best size either restrained or un - \ restrained.



The graph (enclosed, Fig.6) was produced by the calculated brick-work" program and it indicates all possible combinations of height and load for eight combinations of mortar and brick. The program is really a one.off since once all 96 available varieties have been plotted (on 12 sheets of paper for clarity) you never need to run it again (assuming you don't loose the output). These diagrams are shortly to be available to architects via the Brick Development Association.

These structural programs were written quite recent ly and subsequently we haven't really tried building their use into the. curriculum. The few items of software **in environmental engineering however have been** around for quite a long time, especially the daylight program which I originally wrote on the Department's Nova minicomputer. The latest version allows the student to define the dimensions of a rectangular room and position windows on walls (using the joystick naturally). The light levels are then either displayed on the screen or plotted out (see drawing, Fig.7).



This program's used as part of the environmental science teaching in First Year, where students check light levels in their design using the program, the BRS protractors (boring), or the artificial sky (which has to be seen to be believed). The quality of the results and the speed with which they are obtained convinces even the most recalcitrant that it's worth doing.

In this same area Neil Sturrock (head of Environmental Science teaching in the Department) has written a program to calculate heat losses and loads in a structure; and the interstitial condensation risk in multi layer construction - these two are used by students ,in order to check results in lab work

The project to develop educational software is only beginning and we hope to develop a number of properly instructional programs over the next 18 months, as opposed to the rather ad hoc collection of stuff just described.

One of the problems that I am just beginning to face is that *on* the one hand you have people who can write programs and on the other teachers with specialist knowledge. It is very difficult to presuade the latter to set aside the time and generate the energy to sit down tith the former, so as to write a lesson. I am convinced hat in the areas like structures and accoustics, it would be of immense utility not only to the student, but also the members of staff to arrive at some clear and well docume nted tour through certain aspects of the subject but can I convince the other people on the staff? Time will tell.

To end on an even less cheerful note - one of the problems we are beginning to face is the bottleneck of only two machines with two research assistants and 30 tudens all competing for time. We ordered a further

two PETs with the promise of sharing a third which would have solved the problem at least for the next few months, only to find that some mystical figure called 'audit' has stopped all micro purchases in the City. So we wait, and meanwhile write more pleading memos to the relevant committees...

Hardware

The plotter referred to is a Hewlett Packard A4 size '7225A which interfaces directly to the PETs IEEE port. Should you need more information on how to drive plotters from PETs in Basic you could look at the article I've written for IPUG (if they ever publish it).