CS3283 Assignment #2

14th February 2003

Due by 5pm on Friday, March 14, 2003
Delivered to Hugh

You may work in groups of (upto) four students. Do not bother asking for a group of five. The group members do not have to be from your tutorial group, and it is up to you to ensure that everyone contributes equally. Email Hugh with your group members before Friday 21st February 2003.

This assignment is worth 35% of your assignment mark. It consists of the development of a Tcl/Tk program, with some documentation.

Your task...

Your task is to implement and document a Tcl/Tk user interface for a new tool to assist in the management of moveable assets in a building, which I will attempt to describe below.

The interface is a special purpose graphics editor, with some extra features, and is intended to manage the overall process of allocating, editing, finding, moving and querying the moveable assets in a building.

Moveable assets should at least include

- Computers
- Desks
- Chairs
- Filing cabinets
- But ... an extensible application that allows you to add new asset types will get you extr marks...

The user of your application will be using a main screen which shows floor layouts of buildings and is able to add new assets (perhaps using a menu item), and place them at the desired location on the layout.

Each asset may be queried at any time by (perhaps) clicking on it, and a new **info** window will pop up, showing both fixed and editable information about that asset. The fixed information should include:

- The date and time the asset was first entered into the system

- The room it was first allocated to
- The room it is currently in (this changes if you drag-n-drop the asset to a new room)
- A brief description for the asset, created when the asset was first entered into the system

The editable information should include

- A scrollable text box with (say) 5 visible lines of text.

The graphical view displayed should be easy to navigate.

Notes:

The following points form part of the functional specification of the user interface.

1. At any time you should be able to save and restore the state (i.e. the system is persistent)
2. The program should be safe on failure - that is - if some part of your program dies for some reason, you can re-run the program, and get back to where-you-were.
3. The minimum flow of operation of the interface is that you can
   (a) add new assets,
   (b) move existing assets (from room to room), delete assets, copy, paste, and so on,
   (c) edit the **info** asset information,
   (d) save and load new databases,
   (e) Search for an asset, and find out which room it is in
   (f) print out an asset location diagram.
4. You may assume that room boundaries are always rectangular, even if the floorplan is not.
5. For drawing purposes, you can assume that computers would be on-top-of tables, and tables on-top-of the floor.

Tips:

I'm just guessing here, but

1. You probably have to maintain a database file containing the state of the display, and the names, locations and **info** descriptions of the elements in the display.
2. You may have to maintain a configuration file containing other information - such as the name of a graphics file to use as a floorplan, and/or the name of a file containing the room names and locations (remember that your program has to know where rooms are.
3. It may be to your benefit to learn how to use database/configuration files that are themselves written in Tcl/Tk (that is you just use **source filename** to load and use the information in **filename** directly).

Things that will earn you extra marks:

- A useable 3D display
- A floor layout/map editor
- An asset editor
- History maintained (so you can back up to a previous state)

Deliverables:

You are to present your assignment as a single (zipped) file containing the sourcecode and a README file outlining how to run your program, along with an electronic version of the documentation in PDF format. The documentation must also be presented on paper, and should contain:

- A title page containing your names and matriculation numbers.
- Table of contents...
- A one page introduction describing the application function in a brief non-technical style.
- A one to three page section describing the system, (such as how you store the database, how you store the assets, floor plans, file formats and so on). If there are limits on the system, include a brief summary of these constraints.
- A one to three page section describing the interface design (what are the screens, what is the general flow of operation of each screen.
- A user manual for the program

Note that this assignment *does* require you to *implement* the application, and it must be implemented in Tcl/Tk, and runnable either on the cygwin version of Tcl/Tk, or the one on the Suns.

Assessment:

The assessment is as follows:

Documentation	25%
Code style/quality	25%
Operation of the interface	50%

Try to achieve clarity in your writing and take care in the structuring of the document.

COOPERATING VERSUS CHEATING

You are allowed to discuss the problems with your friends, and to study any background material with them, but the assignment *should be your own group's work*. **Copying** and **cheating** will be grounds for failing the assignment.