

**TOPS 2008: Time Management System**  
University of Kent Team

**Rationale:** In any new learning environment, one of the hardest challenges is time management, be that lectures, classes, personal working or social time. With each new assignment or commitment comes an inevitable difficulty in trying to keep on top of them all, leading to missed deadlines and late nights working! Your challenge is to implement a time management system, designed to be used by a new student being inducted into a University.

---

**Base task:** your challenge is to write a time management system for a new University student. This should keep track of all lectures, classes and other such commitments, including social ones. The base system must be able to accept new items, as well as allow for them to be deleted and modified. Don't spend too long on making a complex solution, you will add more functionality in the next part. (20 marks)

**Optional extras:** (N.B. You are not expected to complete anywhere near all of these! It is advisable to tackle one or two and do them well...)

1. In addition to this, the system could be used to keep track of assignments and other pieces of work. Write an add on to handle these sorts of events. (16 marks)
2. Add clash detection to your system to prevent double booking! How you handle this is up to you. (10 marks)
3. Timed reminders: this task involves writing code to allow for timed reminders to be set, notifying the student when a commitment or deadline is approaching. (16 marks)
4. Sound support: to provide audio feedback for the various components of the time management system. (16 marks)
5. Date manipulation: allow not only for a standard date format such as DD/MM/YYYY, but also more user friendly descriptions such as 'next Saturday'. Which date 'strings' you choose to implement is left to your discretion. (16 marks)
6. Load and save to file: in order to simulate the storage and retrieval of data from a remote location, implement a method of saving the file and loading from it. (10 marks)
7. Live updates: implement a method of detecting when a calendar file has been modified on the storage location and retrieve the updates for loading. (20 marks)

8. User interface: implement a simple graphical user interface for all the features you have chosen to implement. (10 marks)

---

Note to markers: it is intended that for each task that half the available marks are for functionality and the other half for code quality.