Reasoning

- **Instance classification and KB consistency.** The answers for the university1.owl exercises can be found at [http://owl.man.ac.uk/2005/07/sssw/university.html](http://owl.man.ac.uk/2005/07/sssw/university.html).

- **Cleaning up a ‘dirty’ ontology.** Let us randomly have a look at a deduction and its explanation (click on the “?” right from the deduction in Protégé) as a first step toward figuring out why so many classes are unsatisfiable (i.e., equivalent to Nothing, or ⊥). Take the explanation for CS.StudentTakingCourses:

This CS.StudentTakingCourses has a long explanation of why it is unsatisfiable, and we see that some of the axioms that it uses to explain the unsatisfiability also have unsatisfiable classes. Hence, it is a good idea to set this aside for a while, as it is a knock-on effect of the others that are unsatisfiable.

Let us have a look at the unsatisfiability regarding departments.
So, the AI Dept is unsatisfiable because its superclass CS Department is, i.e., it is a knock-on effect from CS Department. Does this give sufficient information as to say why CS Department is inconsistent? In fact, it does. See the next screenshot, which is the same as lines 3-7, above.

CS Department is unsatisfiable, because it is affiliatedWith some CS Library that, in turn (by transitivity), is affiliatedWith some EE Library that belongs to the EE Department, which is disjoint from CS Department. Two ‘easy’ options to get rid of this problem are to remove the transitivity or
to remove the disjointness. Alternatively, we could revisit the domain knowledge; e.g., CS library may not be affiliatedWith EE library, but is, adjacentTo or disjoint with the EE library.

Let us now consider why CS\_course is unsatisfiable:

We have again that the real problem is CS\_Department; fix that one, and CS\_course is satisfiable, too.

There is a different issue with AIStudent. From the explanation in the next screenshot, we can see immediately it has something to do with the inconsistency of HCIStudent.

But looking at HCIStudent for a clue does not help us further in isolating the problem:
Considering the axioms in the explanation only, one can argue that the root of the problem is the disjointness between \texttt{AIStudent} and \texttt{HCIStudent}, and remove that axiom to fix it. However, does it really make sense to have the union \texttt{ProfessorInHCI} or \texttt{AI}? Not really, and therefore it would be a better fix to change that one into two separate classes, \texttt{ProfessorInHCI} and \texttt{ProfessorInAI} and have them participating in \texttt{ProfessorInHCI ⊑ ∀ advisorOf HCIStudent} and \texttt{ProfessorInAI ⊑ ∀ advisorOf AIStudent}, respectively.

Last, we have a problem of conflicting cardinalities with \texttt{LecturerTaking4Courses}: it is a subclass of \texttt{TeachingFaculty}, which is restricted to taking at most 3 courses, which is in conflict with the “exactly 4” of \texttt{LecturerTaking4Courses}. This can be fixed by changing the cardinality of either one, or perhaps a lecturer taking 4 courses is not a sub- but a sister-class of \texttt{TeachingFaculty}. 