The Einstein Puzzle

Table of contents

1. Revision history

Date	Description
2005-05-23	Initial version.
2007-01-22	Added the parkinlot problem. Some minor improvements. Note on Wikipedia.

> **Note:**
> The original version of this article was witten on 2005-05-23, and referred to a Wikipedia-Article for the "original" version of the problem and some historic notes. Since then, the said article has been thoroghly rewritten, probably to the better. Among other things, the article has been renamed to `Zebra_Puzzle`, and another formulation is considered "first known publication". Also see the discussion page. I have selected *not* to rewrite the present article to take these changes into account.

2. The Puzzle

The so-called "Einstein's Puzzle" is a well-known logical puzzle. Commonly, it is claimed to have been invented by Albert Einstein, and it is also claimed that he should have stated that only 2 percent of the world's population was able to solve it. There is no authoritative source for these claims. Likely, the claims were invented by the "everything-is-relative"-people. Nevertheless, I will call a puzzle of this sort "an Einstein puzzle" for the rest of this article.

There are many, more or less "isomorphic", versions of "the puzzle". The version in Wikipedia (which I have allowed myself to copy here), goes:

- There are 5 houses. each a different colour.
- A person of a different nationality is in each house.
- The 5 owners each drink a certain drink, each smoke a certain brand of cigarette, and each have a certain pet. No owner has the same pet, smokes the same brand of cigarettes nor drinks the same drink as any other.
- The question is: Who has the fish?

CLUES

- The British man lives in the red house.
- The Swedish man has a dog for a pet.
- The Danish man drinks tea.
- The green house is to the left of the white house.
- The owner of the green house drinks coffee.
- The person that smokes Pall Mall has a bird.
- The owner of the yellow house smokes Dunhill.
- The person that lives in the middle house drinks milk.
- The Norwegian lives in the first house.
- The person that smokes Blend, lives next to the one that has a cat.

- The person that has a horse lives next to the one that smokes Dunhill.
- The one that smokes Bluemaster drinks beer.
- The German smokes Prince.
- The Norwegian lives next to a blue house.
- The person that smokes Blend, has a neighbour that drinks water.

There have been many versions around, in many cases simply with other names of some of the involved properties. More significantly, most versions involve only 14 clues, not 15 as above. With the program presented later, it is easy to show that one rule (but not two!) can be omitted, while still guaranteeing a unique solution.

I was fascinated by this puzzle already as a child, (This is the version that I had) and I felt immensely proud when I succeeded to solve it. I have seen several different versions of such puzzles, some very similar, or even "isomorphic" (equal after a relabeling of properties or property values). This article presents four different Einstein-puzzles, clearly non-isomorphic. They differ in their number of properties (= p) and their values (= v), the type of predicates in the rules. The puzzle "Table" below also uses subsets of values (for example: from the property first_name the sex of the person can be inferred. Some rules state that a certain property-value pair is associated with a certain sex.)

It also deserves to be pointed out, that all of the puzzles originate in verbal form, and the translation to properties, values, and predicates in some case is quite non-trivial, and involves other than pure logical reasoning. In the original problem, it is (somehow?!) obvious that "neighbor" means immediate neighbor, an interpretation that is not the same as the meaning in normal day use (nor the meaning in mathematics). Also, for example, the German wording of the table-puzzle contains clues (in particular regarding the sex of the persons), that for someone with only elementary knowledge of the language may not catch. Even worse, in some cases, cultural interpretations may be implicitly assumed: A person whose mothers tongue is a right-to-left-language may interpret the statement "The Norwegian lives in the first house." different from a left-to-right-"speaker".

We define an "Einstein-problem" to consist of

- p *properties*, each with
- v *values*, together with
- a number of *rules* (predicates) which restrict the possible solutions

As defined, an Einstein-puzzle can have zero, one, or many solutions. Of course, a "good" Einstein-puzzle should have exactly one.

If "position" (in any disguise) is a part of the problem, we will count it to the number of properties, p. With no rules, it is easy to see that the number of solutions is $v!^{(p-1)}$. (The "-1" comes from the fact that one property, which normally can be interpreted as a position, is used to to "index" a solution.) For the original Einstein puzzle, this number amounts to $5!^5 = 2.4883200000 * 10^{13}$.

A rule like "The owner of the green house drinks coffee" (we will call such a rule an

equivalence in the sequel) restricts the number of solutions with a factor of v. A rule like "The person that smokes Blend, lives next to the one that has a cat." is already more complicated to analyze: If the Blend-smoker lives in one of the two edge-houses, this nails the cat-owner as his (only) neighbor, if the Blend-smoker does not live in an edge-house, he has two neighbors, one of them must be the cat-owner. It is easy to come up with some upper and lower estimates for the number of rules necessary for a unique solution, but it is probably at most in trivial special cases possible to come up with criteria for a unique solution. Oh well, we do not know an optimal strategy for chess either... :-)

However, the program presented later can be used to compute the number of solutions for a particular Einstein puzzle. Thus, e.g., for a problem with a unique solution, for every rule it can be tested whether it is necessary for the uniqueness of the solution.

We also remark that the property "position" (possibly in some disguise), as opposed to most other properties, has more structure (for example, it is a totally ordered set, and also has a metric), and therefore concepts like neighbor, facing, preceeding etc make sense, which would not make sense on an arbitrary set.

3. Five Einstein-Puzzles

In the following table, four different, clearly non-isomorphic, Einstein-puzzles are presented. Clicking on the name will open the puzzle description page. The meaning of the XML-column will be explained shortly.

Name	XML	Props	Values	Used Predicates	Subsets	Source	Comment
Einstein	XML	6	5	equivalence neighbor, to_right	-	The original puzzle, according to Wikipedia	The Original! This version has non-minimal rule set (15 rules, while 14 suffice. For example, rule 15 can be deleted.).
Doctors	XML	4	4	exclusion	-	Source: this forum, (contribution from quantumfluf which almost	Uses exclusions exclusively (sorry :-)

						surely is not the original source.	
Table	XML	4	8	equivalence exclusion, neighbor, property_in, gegenueber	yes	From the German magazine "Die ZEITMagaz See scan of the original article.	The verbal formulations are somewhat tricky, containing a good deal of the difficulties, therefore I do not provide a translation, but leave it it in German.
Elskaren	XML	6	10	equivalence facing, exclusion, neighbor, not_neighbo	left_side, right_side	Found in a local student magazine "Elskaren" for/by the electrical engineering students at Lund Institute of Technology Translated from Swedish. Scan.	This is basically a blown-up version of the original puzzle, with ten houses instead of five. Non-minimal rule set (rules 16 and 27 can be eliminated).
Parkinglot	XML	6	5	equivalence neighbor, not_neighbo	-	Found in German student forum www.mathe	Translated from German.

4. Making it formal

To be able to solve a puzzle with computer, we first have to translate it to a formal notation. I have selected to design an XML DTD for this task. Generated documentation for this DTD is found here. Thus, for a, verbally described Einstein puzzle, an XML file,

valid with respect to said DTD, has to be (manually) written. For example, einstein.xml is the XML-File in which the problem in the Wikipedia version is formulated. Actually, the XML-version of the puzzles in the table above, is available through a click in the second column.

We now give an informal description of the semantics of the XML file. The XML file first defines the different properties, and their possible values. The used names have to adhere to the syntax of identifiers in the C language (ascii-letters, digits, and underscore "_"). If necessary, subsets can be defined here, too. Then a number of rules are defined. A rule consists of one or more predicates, predicates being `equivalence`, `exclusion`, and *if* position is among the properties, `neighbor`, `facing`, `gegenueber`, `to_right`, and `not_neighbor`. A predicate takes two or more property-value pairs. So is the rule "The British man lives in the red house" translated into

```
<equivalence>
    <property-value name="nationality" value="english"/>
    <property-value name="color" value="red"/>
</equivalence>
```

For a property-value in a subset, there is the predicate `property_in`.

A peculiarity, that somehow sneaked into the code, is that for the property position, the values are ignored, and instead C-style indexing is used (i.e. starting with 0). For example, the rule "The person that lives in the middle house drinks milk" should be translated into

```
<equivalence>
    <property-value name="position" value="2"/>
    <property-value name="drink" value="milk"/>
</equivalence>
```

The DTD also allows for textual information, as well as pictures.

Later, the XML file is translated into a C++-program, that will solve the puzzle. Also, on this web-site, based on Apache Forrest, HTML-Files, like for example einstein.html, are automatically generated from the XML-files, with no human inputs.

5. Programming

The transformation of the XML-File to a C++ program is carried out by a Metamorphosis script. Metamorphosis is a powerful and generic tree transformation tool from Ovidius. It is a commercial program, but can be downloaded free of charge.

To run, you need Metamorphosis (any version should do), a C++-compiler (I have only tried gcc, but there are no special requirements), and make (preferably). Download the Puzzle Kit from the download section, unpack in an empty directory, make sure Metamorphosis and the C++ compiler works, adjust the paths in the Makefile, and type "make".

There are some command line options, most are only of interest for debugging. (See the code in `generic.cc`.) However, the command line option `-#` makes the program give out *all* possible solutions; the default behavior of the program is to terminate as soon as one solution has been found. This option can be used to determine the number of solutions; in particular to determine the uniqueness of "the" solution.

The execution of most puzzle solvers take only some milliseconds. The "elskaren" puzzle, due to its larger size, takes somewhat longer: With maximal optimization (this makes a huge difference for elskaren) it takes 2.3 seconds on my Athlon 1800XP.

6. Contributions

I would be happy for other "Einstein Puzzles", in particular if they are not isomorphic with any of the herein presented.

> **Note:**
> If you have never compiled C++ programs before, you may need assistance. There is nothing wrong with that, and does not make anyone stupid. However, I kindly request not to be sent questions on how to compile C++ program, etc.

7. Downloads

File	Version	Description
Puzzle Kit	2005-05-27	Complete kit with Metamorphosis script, include file, XML-Files, and Makefile
C++ Files	2005-05-27	The generated C++ Files `einstein.cc`, `elskaren.cc`, `table.cc`, and `doctors.cc`. Only for those without Metamorphosis.

8. Links

This is of course not the only web page that deals with the Einstein puzzle. Here are some links, that was not mentioned earlier.

- http://www.mindspring.com/~mccarthys/puzzle1.htmStraightforward, pen-and-paper type solution.
- http://www.stanford.edu/~laurik/fsmbook/examples/Einstein'sPuzzle.htmlElegant solution based on logical programming.
- http://www.rakkav.com/homeworlds/brainstem/pages/einstein.htmSome interesting discussion, in particular on verbal formulations and ambiguity. And, of course, a solution of the standard puzzle.