

## Second Laboratory Assignment

November 2nd 2009

The goal of this assignment is to develop a knowledge based system using the CLIPS language that solves the following problem.

Due to that the housing market is low, the *public office for house renting* of the *Department of Housing* wants to facilitate the process of renting a house to the citizens, so they can find the house that fits their needs best.

This office has information about house renting all over the country. To simplify our problem we will assume that our system has only to recommend housing rentals for the fictitious city of *AnyWhereTown*. Each offer of housing rental includes the most relevant characteristics of the house, for example:

- Monthly rent in euros.
- Habitable space in m<sup>2</sup>.
- Number of bedrooms and if they are single or double bedrooms.
- If the house has a terrace or an small balcony.
- If the house is furnished and/or with appliances.
- Kind of housing: flat, duplex apartment, detached/semidetached house. In the case of flat or duplex, the floor (ground level, first floor, second floor, ..., attic).
- Other characteristics as: Pets do not allowed, heating or air conditioning, if it is oriented to the sun (in the mornings, in the evenings, all day), access to common swimming pool, parking space included, with excellent views, ...

You must include more characteristics to complete the description of the offers. You can find an exhaustive list of characteristics in any web site for housing rental advertising, for example [www.idealista.com](http://www.idealista.com).

Additionally to these characteristics, the system has to use the knowledge about the city to deduce other characteristics that could be important to the clients. For example, if there are near services of special interest, such as public transportation (subway/bus stops), shopping areas, supermarkets, hypermarkets, malls, schools, health services, green areas, ... To be near to some of this services/areas could be annoying for some clients or desirable for others, such as nightlife areas, stadiums, ...

To determine this kind of information we have a list of all the services/areas of the city and their coordinates. We will also have the coordinates of each renting offer. The coordinates of the city are measured from an origin and they tell the relative position in two dimensions of any place in the city to that origin measured in meters. We can determine if an specific place is near, mid distance or far to another depending on if it is less than 500 meters, less than 1000 meters or more that 1000 meters respectively.

People looking to rent a house have constraints and/or preferences about the kind of space they want to rent. We also know their personal characteristics, for example:

- Maximum rent they want to pay and if this is a hard constraint or they would pay more if the place is good.
- Constraints about the number/sizes of the bedrooms.
- Minimum rent that for the client indicates that the offer is too cheap to be true (the place is probably a dump).
- Specific constraints or preferences about the distance to an specific service (near to schools, near to public transportation, ...).
- If the client prefers public transportation to move inside the city.
- Age of the people looking to rent.
- Typology of the people looking to rent: couple without children, couple that will have a baby in the near future, family (mono parental, bi parental) with children/elder members (number), group of students (number), just one person, ...
- If they work/study in the city and where is their place of work/study.
- If the client owns a car.

You must include more constraints/preferences/characteristics.

This characteristics determine, depending on the profile of the tenant-to-be, that some services are more important than others, for example:

- To young people could be more important to have nightlife areas near.
- To families and couples could be more important to have a hypermarket near in order to do their monthly shopping.
- To elder members of the family is more important to have near health services or supermarkets for their daily shopping.
- If there are children, it could be interesting to have near or at mid distance schools and green areas.
- It could be also important to have near public transportation to go to work/study.

There are also other common sense characteristics that make a place better than other, like being a sunny place, that it is better a flat that is an attic than a ground level flat, that a detached/semi-detached house is better for larger groups of people, that is good to have green areas not far, areas with more services around are better (the area has more life), that are better the areas that have no services considered annoying, ...

You have to code all this knowledge in your system.

A solution will be a list of the more adequate offers for a client, giving also a recommendation level:

- Partially adequate: Only one or two criteria fail, but it could be acceptable depending on client consideration (rent is a little bit more than the specified maximum, public transportation is at mid distance, there are not green areas at near distance, ...).
- Adequate: All requirements are fulfilled by the offer.
- Very recommendable: All requirements are fulfilled and it has extra characteristics that make this offer better than the rest (lower rent, more rooms, more services near than the explicitly specified, ...).

For the offers that are partially adequate the criteria that fail have to be specified. For the offers that are very recommendable it has to be specified what characteristics make that offer better than the rest.

You have to implement this system and write a report about it that has to include:

- An analysis of the problem **following** the steps of the knowledge engineering methodology explained in class (identification, conceptualization, formalization, implementation and testing). You have to detail this steps in your documentation. Think that the conceptualization and formalization steps include the ontology, so, you have to explain with detail how it has been designed.
- The ontology created to represent the knowledge that the system needs (you can generate the description and the graphical representation for the documentation using Protégè)
- Commented source code of the implemented system.
- A set of non trivial tests that demonstrates how your program solves different cases.

The grade will depend on the quality of the analysis of the problem, the adequacy of the coverage of the ontology and the quality of the solutions that can be obtained. Read the published evaluation rubrics for a detailed description of the grading criteria.

The written report and the code are due by **January 4th**, the delivery will be electronically following the instructions that will appear in the racó.

This assignment has to be solved preferably by teams of two students.

It is important to plan the development of this assignment and do not do all the work at the last moment. The assignment guide will give you directions about how to develop the assignment and will help you to plan your work. You can find also information or ask question on the assignment electronic forum.