

EECS 3401 Introduction to AI & Logic Programming

Example Specification

Due: Monday, September 29, 5:30 pm

Where: In class

If the class has begun your report is late

1 Main points

Be sure to read and follow all the guidelines from the links on **reports** and **academic honesty** from the WWW home page for the course.

You should read the course web page on reports – link *Reports* in the table of contents.

You should read the notes on testing – link is in the *Resources* page for the course. Of particular use for the Prolog predicates for the first two reports are the notes on partition testing.

1.2 To hand in

1. The first page of the report is a cover page that you download from the forum message with the subject *Example specification*.
2. The body part of the report is a listing of the file *report_example.pl* that is to contain your answers for all the exercises in same order as they are in the specification.
3. For listings a 10 point fixed width font such as courier should be used. Note that cutting and pasting program text into an MSword file does not comprise a listing; a listing is an exact print out of the program text file.

Please use the unix command **enscript** to print the file *report_example.pl*.

```
enscript report_example.pl
```

I upload the file from my Mac to Prism and use the following command that produces the Postscript file *report_example.ps*. I then download the Postscript file and print it on my home printer.

```
enscript -p report_example.ps report_example.pl
```

1.2 Electronic submission

Before the deadline, submit the report file. No other files should be submitted. You should be submitting the file after completing an exercise every few days so you always can show that you have done some work – the closer to the deadline, the more work should be done. Leaving work to the last day is a poor and inefficient work and learning strategy, and of course there could be network or computer problems at the last moment that could prevent you from completing your report.

To submit your report use the following command on Prism.

```
submit 3401 re report_example.pl
```

Files cannot be deleted – the submit command can only add or replace files – so be very careful to not submit useless files.

While you can develop your programs on your personal computer, be sure your files will load and execute correctly on Prism.

2 The Exercises

Exercise 1

Document and explain the execution trace of the query shown in the following. Clearly show the values of all variables at all times, including when they are renamed. Back substitute the variables until you know the first value for **S**. For an example and style to follow, look at the trace for **append** in the utility slides. **You must do this manually** without using Prolog.

Trace the following suffix predicate with the query **suffix(S, [1,,2,3])**.

```
suffix(S, [_ | L]) :- suffix(S, L).
suffix(L, L).
```

Exercise 2

1. Define the predicate **prefix(Prefix, List)** that asserts that **P** is a prefix of the list **List**. The definition only needs the predicate **prefix/2**.
2. Define the predicate **slAppend(Sublist, List)** that asserts **Sublist** is a sub-list of the list **List**. For this definition you are to use **append**.
3. Define the predicate **slPrefix(Sublist, List)** that asserts **Sublist** is a sub-list of the list **List**. For this definition use the predicate **prefix** from Exercise 2.1.

Exercise 3

Define the predicate **total(NumberList, Total)** that asserts **Total** is the sum of all the numbers in **NumberList**.

Exercise 4

Define the predicate **strange_swap(Numbers_1, Numbers_2, Smaller, Bigger)** that asserts given **Numbers_1** and **Numbers_2** are equal length lists of numbers, then for numbers in the corresponding positions of the list, the smaller number is in **Smaller** and the larger is in **Larger**.

```
#Numbers_1 = #Numbers_2
∀j : 1 .. #Numbers_1 | Numbers_1[j] ≤ Numbers_2[j]
    • Smaller[j] = Numbers_1[j] ∧ Bigger[j] = Numbers_2[j]
∀j : 1 .. #Numbers_1 | Numbers_1[j] > Numbers_2[j]
    • Smaller[j] = Numbers_2[j] ∧ Bigger[j] = Numbers_1[j]
```