Introduction to Turbo Prolog

Xindong Wu

1 Getting Started

To download a copy of the Turbo Prolog system to your soft disc (or a Prolog directory on your PC hard disc):

```
ftp kais.emba.uvm.edu
login as anonymous
cd prolog
bin
prompt
mget *
```

To run Turbo Prolog, issue the following command:

`Prolog`

Press the space bar after the prompt information, and you will come up with the user interface of the Prolog system. The main menu is displayed on the top of the interface, and there are 4 windows in the middle. The sizes of the 4 windows can be changed by selecting the **S**etup/**W**indowsize option in the main menu. The **S**etup/**S**ave configration option can be selected to save your changes, in the defalut file `prolog.sys`.

Help information (such as function keys) is always given on the bottom line.

1.1 Example 1

Edit, compile and execute the following program with Prolog:

```
predicates
  hello
goal
  hello.
clauses
  hello :- makewindow(1,7,7,"My first program in Prolog",
                      4,56,10,22),
           nl,write("Please type your name: "),
           readln(Name),nl,
           write("Welcome ", Name, "!"),nl,
           readln(_).
```

Editing: Move the cursor to **E**dit in the main menu and press the return key, or press the `E` key directly. The editing window is now highlighted and you can start to edit your program. Use the arrows to move the cursor, and `<Del>` and `<Back Space>` to delete letters forward and backward.

Execution: Press `<Esc>` to quit editing, and choose **R**un from the main menu. If the above program has been correctly edited, you will be asked to input your name and answered with the welcome information. If your program is not yet correct, you can `<Esc>` first to get back to the main menu, **E**dit the program, and **R**un the program again. If your program runs into endless loops, `<Ctrl>-<Break>` to quit the execution.

File Processing: To save the program you have just edited, select **F**iles in the main menu, choose **D**irectory under **F**iles to specify a driver and (sub)directory, and **S**ave the program in the **F**ilename. You can input myfirst as your file name and the Prolog system will automatically append an extension `.pro` for you.

1.2 The Editor

The editor embedded in Turbo Prolog is a full screen text editor. Commands with this editor are similar to those in WordStar, Multimate and the Turbo Pascal editor.

Edit	start editing (from the main menu)
`<Esc>` or `<F10>`	quit
`<Del>`	delete forward
`<Back Space>`	delete backward
`<PgUp>`	move the cursor backward one page
`<PgDn>`	move the cursor forward one page
`<Home>`	move the cursor to the beginning of the current line
`<End>`	move the cursor to the end of the current line
Files/**S**ave	save the contents in the editing window
Files/**L**oad	load a file into the editing window
`<Ctr>`-K-B	label the beginning of a block
`<Ctr>`-K-K	label the end of a block
`<Ctr>`-K-H	delabel the currently labeled block
`<Ctr>`-K-C or `<F5>`	copy the labeled block to where the cursor is located
`<Ctr>`-K-V or `<F6>`	move the labeled block to where the cursor is located
`<Ctr>`-K-Y or `<F7>`	delete the currently labeled block
`<F3>`	search a string (globally or just once)
`<Shift>-<F3>`	repeat the search
`<F4>`	search and substitute a string(globally or just once)
`<Shift>-<F4>`	repeat the search and substitution

2 Program Structure

2.1 Example 2

```
          /* Program 2 */
domains
  person,activity=symbol
predicates
  likes(person,activity)
clauses
  likes(ellen,tennis).
  likes(john,football).
  likes(tom,baseball).
  likes(eric,swimming).
  likes(mark,tennis).
  likes(bill,X) :- likes(tom,X).
```

2.2 Example runs

Run the above program, you will get the following prompt in the dialogue window `Goal:_`

Belows records some example inputs and their corresponding responses from the Prolog system. Words in roman are input from the user.

`Goal:` likes(bill, baseball).
`True`
`Goal:` likes(tom,baseball).
`True`
`Goal:` likes(bill, tennis).
`False`
`Goal:_`

`<F8>` to repeat your `Goal` in the dialogue window (for editing or execution).

2.3 Program structure

```
domains
  /* to declare the type of each name to be used in predicates */
predicates
  /* to declare each predicate and the types of its objects */
goal
  /* This can be one or more of the predicates with constants and
     variables as their objects. */
clauses
  /* to define the predicates */
```