

**School of Computer Science and Engineering**

**CZ3005 - Artificial Intelligence**

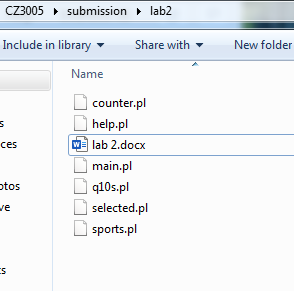
Lab 2 TSP4

Done By:

|  |  |
| --- | --- |
| Ching Jia Chin | U1620237E |

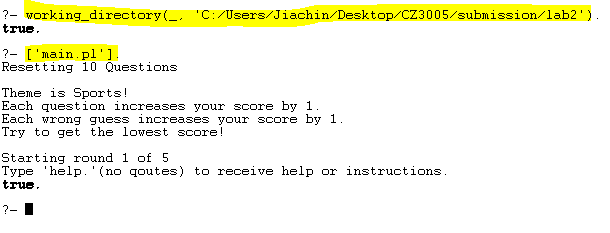
Setup:

1. Add all files, “counter.pl, help.pl, main.pl, q10s.pl, selected.pl and sports.pl”, to a folder.



Files inside lab2 folder

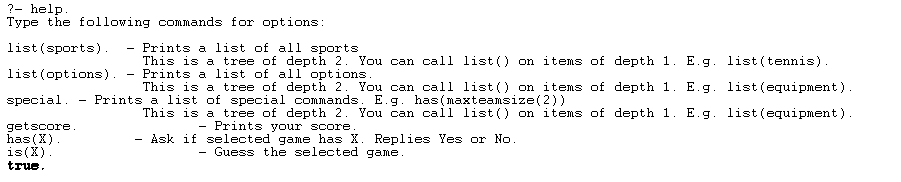
1. Inside Prolog, change working directory to folder.
2. Load main.pl. CMD is “[‘main.pl’].”



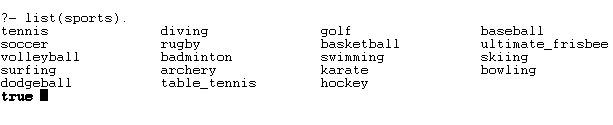
Loading main.pl

How to play:

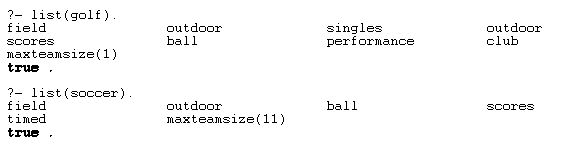
1. Type “help.” to see available commands.



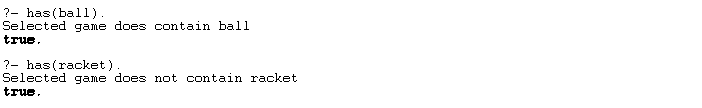
* 1. Type “list(sports).” To see a list of sports.



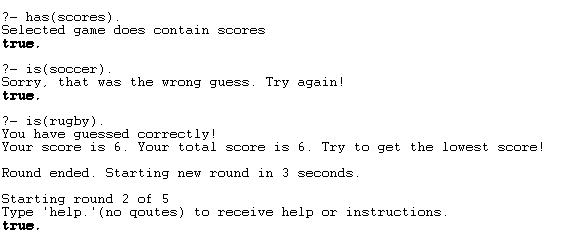
* 1. Type “list(soccer)” to see the contents of soccer.



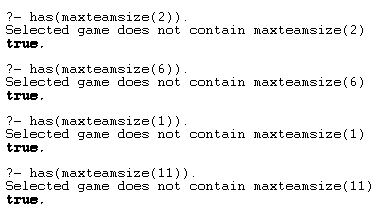
1. Use “has(myquestion).” to ask a question.



1. Use “is(myguess).” to guess the answer.



* 1. Use “Has(maxteamsize(2))” to ask if game has max team size of a 2.





How it works:

Counters

Since prolog is a declarative language, it does not have variables. Instead, counters must be implemented in a declarative way. How I implemented counters is by declaring counters as 1 initially. Every time I wish to increment counter, I must increment 1 to 2, retract previous declaration that counter equals 1, and declare counter as 2. This repeats each time increment is called. The other counters such as rounds and scores are implemented the same way.

Has()

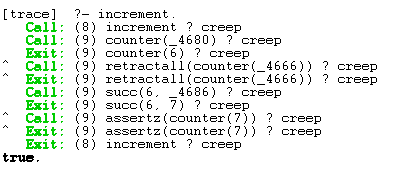
Has(X) was implemented by comparing X with every item in the selected sport. If X matches an item, it will return true. This is done in Prolog by recursively checking every item in a list in sequence. If the end of the list is reached without finding a match, it returns false.

Is()

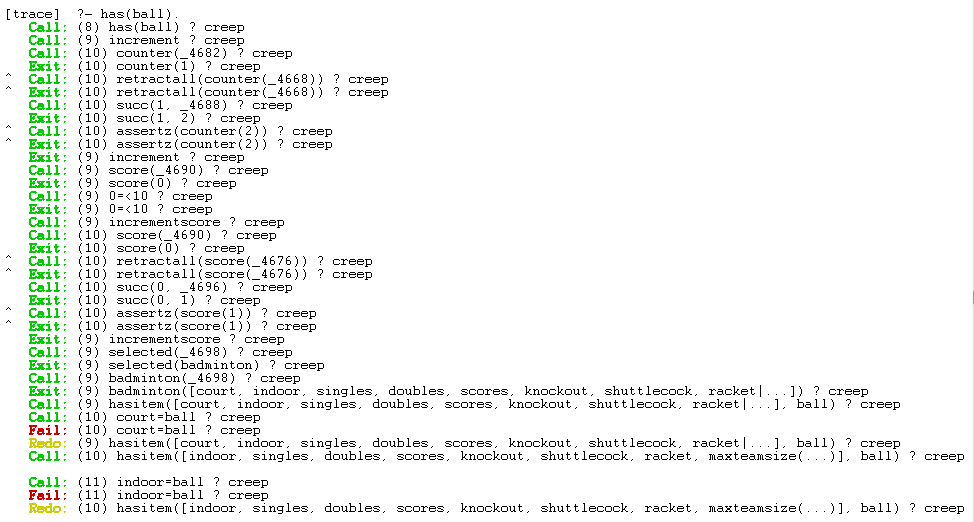
Is(X) was implemented by comparing X with the selected game. If they are the same, return true.

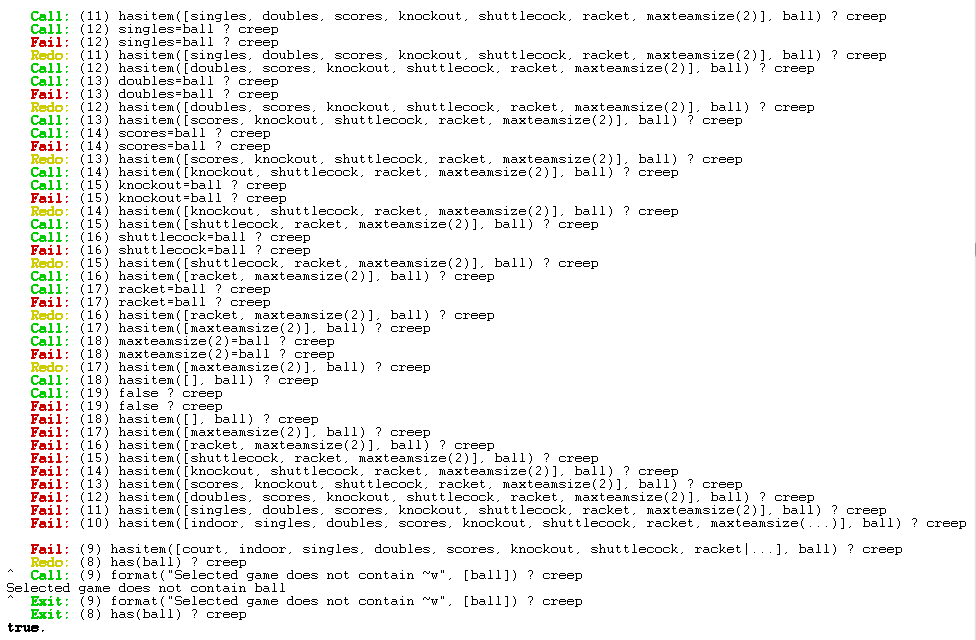
Traces:

Counter()

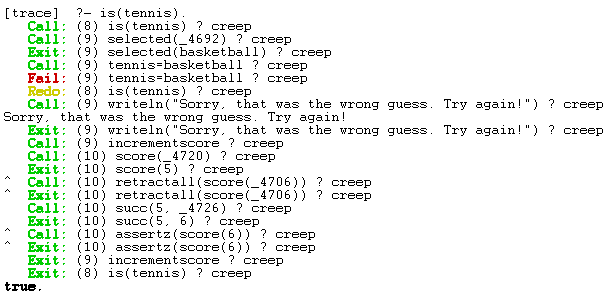


Has()

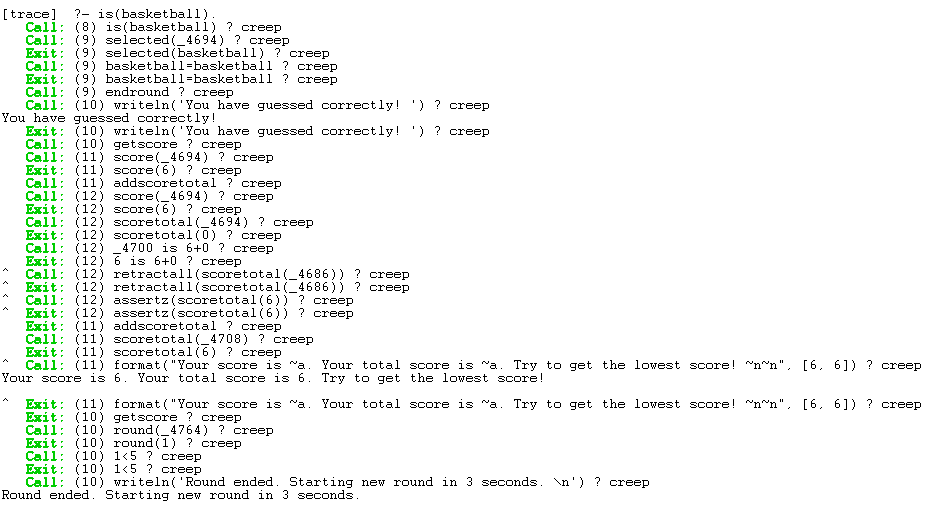




Is()



Failed Is()



Successful is()