Homework 3 - Prolog

Goldbach's conjecture

Every even integer greater than 2 can be expressed as the sum of two primes.

Best-known unsolved problem in number theory.
To this day, the conjecture has been shown to hold up through **4×10^{18}**.

Input

- One even integer greater than 2.

Output

- Two prime integers and their sum is the input integer.
- The output should be ordered.

Example

```
Input: 4
Output: 2 2
Input: 100
Output: 47 53
```

SWI-Prolog

Version: 7.2.3

SWI-Prolog Office Website:
http://www.swi-prolog.org/

Install SWI-Prolog on Windows

Go to this website,

http://www.swi-prolog.org/download/stable

32 bit:
SWI-Prolog 7.2.3 for Windows XP/Vista/7/8

64 bit:
SWI-Prolog 7.2.3 for Windows XP/Vista/7/8 64-bit edition

Install SWI-Prolog on Windows

Add the location of swipl binary file to your PATH variable.

For example, I install swi-prolog in "`C:\Program Files\swipl`".

I should add the path "`C:\Program Files\swipl\bin`" to my PATH variable.

Install SWI-Prolog on Linux

use APT

```
$ sudo apt-get install swi-prolog
```

use YUM

```
$ sudo yum install swi-prolog
```

others

```
Not my business...:)
```

Install SWI-Prolog on OS X

use homebrew

```
$ brew install swi-prolog
```

use macports

```
$ sudo port install swi-prolog
```

If you never believe anyone,

Compile by yourself...

SWI-Prolog source for 7.2.3

Command Line Interface

for Windows

命名提示字元

for Unix-like OS

終端機

Interactive Env

```
$ swipl
```

Interactive Env

```
$ swipl
Welcome to SWI-Prolog (Multi-threaded, 64 bits, Version
6.2.6)
Copyright (c) 1990-2012 University of Amsterdam, VU
Amsterdam
SWI-Prolog comes with ABSOLUTELY NO WARRANTY. This is free
software,
and you are welcome to redistribute it under certain
conditions.
Please visit http://www.swi-prolog.org for details.

For help, use ?- help(Topic). or ?- apropos(Word).

?-
```

```
-? X is 3 + 2 * 7.
```

```
-? X is 3 + 2 * 7.
X = 17.

-?
```

```
-? X is 3 + 2 * 7.
X = 17.

-? wirteln('胡老師最正了').
```

```
-? X is 3 + 2 * 7.
X = 17.

-? wirteln('胡老師最正了').
```

```
-? X is 3 + 2 * 7.
X = 17.

-? wirteln('胡老師最正了').
胡老師最正了
true.

-?
```

```
-? X is 3 + 2 * 7.
X = 17.

-? wirteln('胡老師最正了').
胡老師最正了
true.

-? halt.
```

```
-? X is 3 + 2 * 7.
X = 17.

-? wirteln('胡老師最正了').
胡老師最正了
true.

-? halt.
% halt
$
```

Prolog File

```
1   % filename: sum.pl
2   sum([],0).
3   sum([H|T],S) :- sum(T,S1), S is S1 + H.
4
5
6
7
8
9
10
11
12
13
14
```

```
$ swipl -q
-? [sum].
```

```
$ swipl -q
-? [sum].
% sum compiled 0.00 sec, 3 clauses
true.

-?
```

```
$ swipl -q
-? [sum].
% sum compiled 0.00 sec, 3 clauses
true.

-? sum([1,2,3,4], X).
```

```
$ swipl -q
-? [sum].
% sum compiled 0.00 sec, 3 clauses
true.

-? sum([1,2,3,4], X).
X = 10.

-?
```

```
$ swipl -q
-? [sum].
% sum compiled 0.00 sec, 3 clauses
true.

-? sum([1,2,3,4], X).
X = 10.

-? halt.
```

```
$ swipl -q
-? [sum].
% sum compiled 0.00 sec, 3 clauses
true.

-? sum([1,2,3,4], X).
X = 10.

-? halt.
% halt
$
```

Script File

```
1   % filename: hanoi.pl
2   move(1, X, Y, _) :-
3           write('Move top disk from '), write(X), write('
4   to '), write(Y), nl.
5
6   move(N, X, Y, Z) :-
7           N > 1, M is N - 1, move(M, X, Z, Y), move(1, X,
8   Y, _), move(M, Z, Y, X).
9
10  main :-
11          move(3, left, right, center), halt.
12
13  :- initialization(main).
14
```

```
$ swipl -q -s hanoi.pl
```

```
$ swipl -q -s hanoi.pl
Move top disk from left to right
Move top disk from left to center
Move top disk from right to center
Move top disk from left to right
Move top disk from center to left
Move top disk from center to right
Move top disk from left to right

$
```

Deadline

2016/05/08 22:00

Any Question?

If you still have any question about SBCL,

Read **T**he **F**riendly **M**anual.

Prolog Tutorial:

http://www.learnprolognow.org/

http://www.csupomona.edu/~jrfisher/www/prolog_tutorial/contents.html

SWI-Prolog Document:

http://www.swi-prolog.org/pldoc/refman/

Read **T**he **F**riendly **M**anual or
Use **T**he **F**riendly **G**oogle first,

before you ask teacher or TAs.

Thanks and good luck :)