Prolog Tutorial 2

Question 1:
Using the predicates:

input1(G,I)   input 1 of gate G is I
input2(G,I)   input 2 of gate G is I
output(G,O)   output of gate G is O
andgate(G)   G is an and-gate
orgate(G)   G is an or-gate
xorgate(G)   G is an exclusive-or-gate
and any others you want

write a Prolog program that describes how and/or/xor gates work in general. Also describes the configuration of gates in the figure below. Your program should allow queries about outputs of any of the gates. Test the program for all the gates.

An and-gate produces 1 if both of its inputs are 1, otherwise it produces 0.
An or-gate produces 0 if both of its inputs are 0, otherwise it produces 1.
An exclusive-or-gate produces 1 if its two inputs are different, otherwise it produces 0.

Question 2:
The Fibonacci sequence[1]   f(1), f(2), f(3),.. is:
1, 1, 2, 3, 5, 8, 13, 21, 34, 55......

As you see the definition is easy to grasp:

f(1) = f(2)= 1
f(n) = f(n-2) + f(n-1), if n >= 3

Define a Prolog predicate fib(N, R) that computes the Nth number in the Fibonacci sequence.

Example query:
?- fib(6,R).
R = 8

---

[1] The Fibonacci sequence first appears in a book in 1202 by Fibonacci. He considers an idealised growth of a rabbit population, with the following assumptions:
A newly born pair of rabbits, one male, one female, are put in a field.
Rabbits always mate at the age of one month so that at the end of its second month a female produces another pair of rabbits.
Rabbits never die and a mating pair always produces one new pair (one male, one female) every month from the second month on. So f(N) is the number of rabbits at the end of the Nth month.
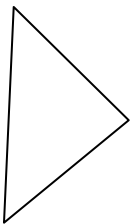These numbers turn up in unexpected places in nature and even in computing!

1

0

1

g1

g2

g3

g4

g5
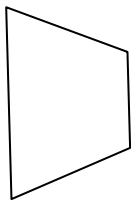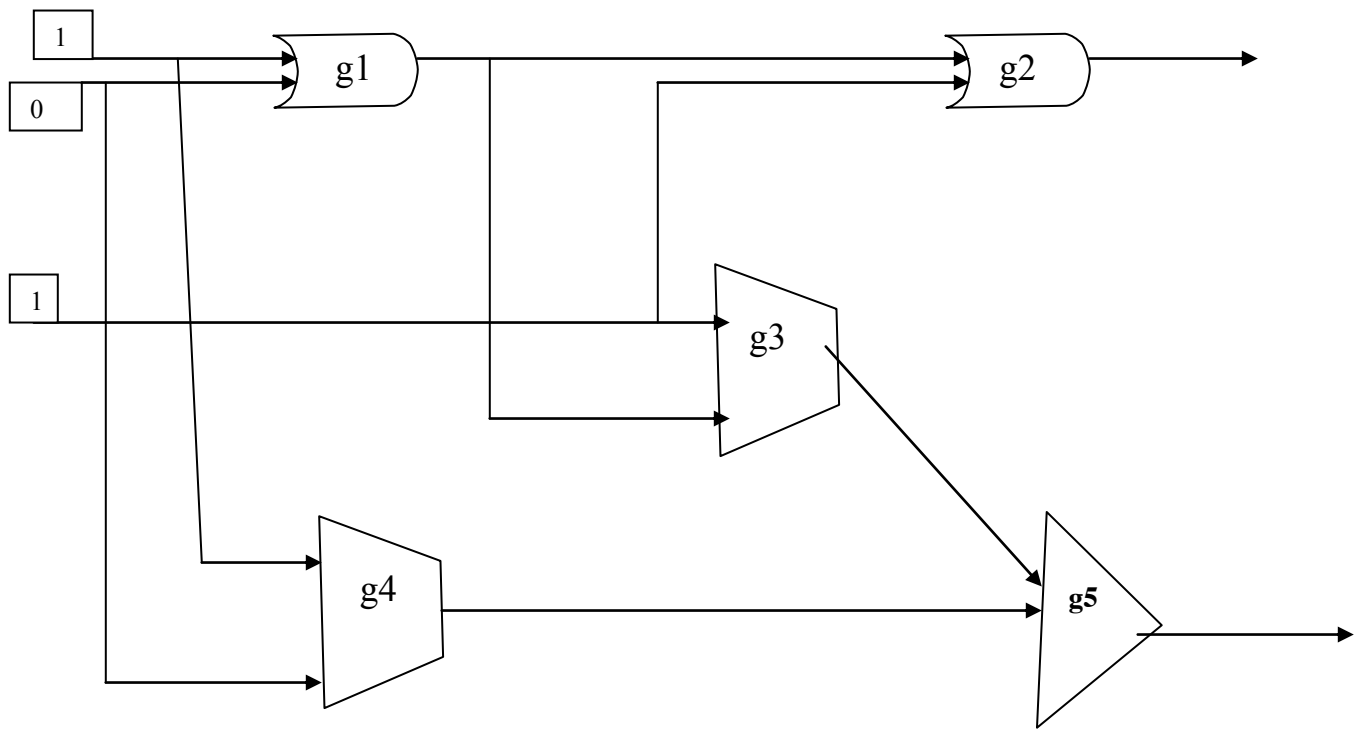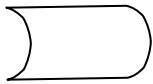
Denotes
And-gate

Denotes
Xor-gate

Denotes
or-gate