Problem 1

Two fish swim in a pond. One is a piranha and the other is either a clownfish with probability 0.5 or another piranha with probability 0.5.

1. A fisherman takes a fish from the pond. What is the probability that it is Nemo?
2. The fisherman observes it is a piranha. What is the probability that the other fish is Nemo?
3. Write a probabilistic program (over the simple probabilistic language) that encodes subproblem 2.
4. Show the probabilistic semantics of the program.

Solution (1)

Define a random variable t for the type of the fish

The probability is

$$P(t = Nemo) = P(\text{catches fish 1}) \times P(t = Nemo|\text{fish caught is 1}) + P(\text{catches fish 2}) \times P(t = Nemo|\text{fish caught is 2}) = 0.5 \times 0 + 0.5 \times 0.5 = 0.25$$

Solution (2)

Define a random variable o for the type of the other fish

Using Bayes rule:

$$P(o = Nemo|t = Piranha)$$

$$= \frac{P(t = Piranha|o = Nemo) \cdot P(o = Nemo)}{P(t = Piranha)} = \frac{1 \cdot (0.5 \cdot 0.5)}{1 - 0.25}$$

$$= \frac{0.25}{0.75} = \frac{1}{3}$$

Solution (3)

```
x₁ := 0;
```

x_1 := 0;

x_2 := Bern(1/2);

$(t, o) \coloneqq (0{,}0)$;

if Bern(1/2)

  t = x_1; $o = x_2$;

else

  $t = x_2$; $o = x_1$;

observe(t $= 1$);

return o;

Solution (4)

The state consists of the four (random) variables: x_1, x_2, t, o

State also contains the probability of a statement to be in this state

The initial state:

$$((x_1 : \bot, x_2 : \bot, t : \bot, o : \bot), 1)$$

Solution (4)

$((x_1 : \bot, x_2 : \bot, t : \bot, o : \bot), 1)$

x_1 := 0;

$((x_1 : 0, x_2 : \bot, t : \bot, o : \bot), 1)$

x_2 := Bern(1/2);

$((x_1 : 0, x_2 : 0, t : \bot, o : \bot), 0.5)$ $((x_1 : 0, x_2 : 1, t : \bot, o : \bot), 0.5)$

$(t, o) \coloneqq (0,0)$;

$((x_1 : 0, x_2 : 0, t : 0, o : 0), 0.5)$ $((x_1 : 0, x_2 : 1, t : 0, o : 0), 0.5)$

if Bern(1/2)

t $= x_1; o = x_2;$

$((x_1 : 0, x_2 : 0, t : 0, o : 0), 0.25)$ $((x_1 : 0, x_2 : 1, t : 0, o : 1), 0.25)$

else

$t = x_2; o = x_1;$

$((x_1 : 0, x_2 : 0, t : 0, o : 0), 0.25)$ $((x_1 : 0, x_2 : 0, t : 1, o : 0), 0.25)$

observe(t $= 0$);

return o;

$((x_1 : 0, x_2 : 0, t : 0, o : 0), 0.25)$ $((x_1 : 0, x_2 : 0, t : 0, o : 1), 0.25)$

$((x_1 : 0, x_2 : 1, t : 0, o : 0), 0.25)$

Problem 2: The Monty Hall Problem

There are three doors. Behind two of them there are goats, behind the other one there is a car. Monty Hall tells you to pick a door. If you pick the one with the car, it is yours. You pick door 1. Now Monty Hall has to pick a different door that behind which there is a goat. He picks door 3. Now he offers you to switch to door 2.

1. Determine whether you should keep door 1 or switch to door 2. What is the winning probability in each of the cases?
2. Write a probabilistic program (over the simple probabilistic language) that encodes this problem.

Solution (1)

Define random variables:

- d_1, d_2, d_3 encoding that there is a car behind doors 1, 2, and 3
- p that encodes the door you picked
  - p and d_i are independent
- m that encodes the door Monty picked

By the game rules, Monty Hall has to pick a door different from yours, and show a goat. Given Monty Hall’s choice, we can infer:

$P(m = 3|d_3, p = 1) = 0$ (no car behind door 3, since he picked it)

$P(m = 3|d_2, p = 1) = 1$ (if the car is behind door 2, he has to pick door 3)

$P(m = 3|d_1, p = 1) = 0.5$ (if the car is behind door 1, he can pick door 2 or 3)

Solution (1)

The probability that behind door 1 there is a car is given by:

$$P(d_1 = Car|m = 3, p = 1) = \frac{P(m = 3|d_1 = Car, p = 1) \cdot P(d_1 = Car \cap p = 1)}{P(m = 3 \cap p = 1)} =$$

$$\frac{P(m = 3|d_1 = Car, p = 1) \cdot P(d_1 = Car \cap p = 1)}{P(m = 3|d_1, p = 1) \cdot P(d_1 \cap p = 1) + P(m = 3|d_2, p = 1) \cdot P(d_2 \cap p = 1) + P(m = 3|d_3, p = 1) \cdot P(d_3 \cap p = 1)} =$$

$$\frac{0.5 \cdot P(d_1 = Car \cap p = 1)}{0.5 \cdot P(d_1 \cap p = 1) + 1 \cdot P(d_2 \cap p = 1)} = \frac{0.5}{1.5} = \frac{1}{3}$$

$$P(d_2 = Car|m = 3, p = 1) = 1 - P(d_1 = Car|m = 3, p = 1) - P(d_3 = Car|m = 3, p = 1) = 1 - \frac{1}{3} - 0 = \frac{2}{3}$$

You should pick door 2!

Solution (2)

```
doors := [0,0,0];
prize := uniformInt(0,2);
doors[prize] = 1;
choice := uniformInt(0,2);
open := (0 + 1 + 2) - choice - prize;
if choice == prize
        open := (prize + uniformInt(1,2)) % 3;
return Expectation(doors[choice])<Expectation(doors[other]);
```