Discrete Structures (W2): Sets and Predicates

Kalvis
RBS

You will do more experiments with Boolean expressions and start to prove propositional tautologies in Coq. You will also formulate simple "human-oriented" proofs about integer arithmetic and other topics.

Keywords: Coq, Boolean expressions, Syntax trees, tautologies, proofs.

(W2.1) Try Boolean short-circuit evaluation on Python: Define two functions: `f(x)` and `g(x)`. Each of them prints a message and then returns value `False`. For example:

```
def f():
    print('Running f(x)')
return False
```

After that evaluate this statment:

```
if f(x) and g(x):
    print('Unreachable statement')
```

There should be also eager

(W2.2) Compare Boolean expressions in Coq, using prefix and infix notation.

```
Require Import Bool.
Eval compute in orb true false.
Eval compute in true || false.
Eval compute in andb true false.
Eval compute in true && false.
Eval compute in negb false.
Eval compute in if true then 3 else 4.
Definition a := true.
Eval compute in orb a (negb a).

Definition eMiddle (a:bool): bool :=
  orb a (negb a).
Eval compute in eMiddle true.

Definition Nor (a b:bool): bool :=
  negb (orb a b).
Eval compute in Nor true true.
```

(W2.3) Use precedence and associativity to draw abstract syntax trees. Use boolean commands "orb", "andb", "implb" and "negb" to compute in prefix notation this function:

$$E(a,b,c) := a \lor \neg(b \land (a \to c)).$$

Draw the syntax tree of this operation. Its leaves are variables a, b, c (and also constants, if they are present in the expression). Inner nodes are all the Boolean operations with 1 or 2 variables. So, in this tree every inner node has 1 or 2 children.

(W2.4) Read a definition of a DNF (Disjunctive Normal Form). Create a DNF for a 3-argument Boolean function. You can pick the 8 truth falues at random.

(W2.5) Prove that for any odd integer k, the square k^2 gives remainder 1, when divided by 8.

(W2.6) Prove that a positive integer n has odd number of positive divisors (including 1 and n itself) if and only if n is a full square - can be expressed as k^2.

(W2.7) Prove that for any prime number p, the square root $\sqrt{p}$ is irrational.

(W2.8) Prove that for any real number x its rounded value to the nearest tenth (rounding to one decimal place) is equal to $\frac{1}{10}\lfloor 10x + 0.5 \rfloor$.

(W2.9) Prove that there is a function $f : (-\pi;\pi) \to \mathbb{R}$ mapping interval $(-\pi;\pi)$ to $\mathbb{R}$ such that every real number y has exactly one x such that $f(x) = y$.

(W2.10) Prove that it is impossible to enumerate all subsets of natural numbers with natural numbers. (Natural numbers are all integers ≥ 0).

(W2.11) Try out the examples in https://bit.ly/2sfIrUK. Discuss, if you are unsure, how to write proofs. Visit tutorials https://bit.ly/37Zjso0, if you need more inspiration.

(W2.12) Try out the Coq assignment on https://bit.ly/37Zjso0 (the link "Assignment about Coq (1 of 5)").

Some Answers

(W2.1)

Full Python program looks like this:

```
def f(): print('Runs f()'); return False
def g(): print('Runs g()'); return False
def h(): print('Runs h()'); return True

## Prints 'Runs f()':
if f() and g(): print('Unreachable')
## Prints 'Runs f()', 'Runs g()':
if f() & g(): print('Unreachable')
## Prints 'Runs h()' and 'Hi':
if h() or g(): print('Hi')
## Prints 'Runs h()', 'Runs g()' and 'Hi'
if h() | g(): print('Hi')
```

(W2.5) Assume that n is odd. Represent it as $n = 2k+1$. Then

$$n^2 = (2k+1)^2 = 4k^2 + 4k + 1 = 4k(k+1) + 1.$$

Consider two cases:
(1) If k is odd, then $k+1$ is even and $4 \times (k+1)$ is divisible by 8.
(2) If k is even, then $4 \times k$ is divisible by 8.
In either case $4k(k+1)$ is divisible by 8. Therefore $4k(k+1)+1$ always gives remainder 1 when divided by 8.

(W2.6) Biconditional (if-and-only-if) statement contains two implications:
(1) First, assume that n is a full square: $n = k^2$. We need to prove that it has odd number of positive divisors. For any divisor of n: $d < \sqrt{n} = k$ there exists another divisor $d' = n/d$ which is bigger than $\sqrt{n}$ – you can divide all divisors in pairs. In addition, n has a divisor $k = \sqrt{n}$ that is paired with itself.
Example: The number $36 = 6^2$ has these divisor pairs:

$$(1;36),\ (2;18),\ (3;12),\ (4;9),\ (6;6).$$

Since the divisor $\sqrt{n}$ is paired with itself, there is an odd number of positive divisors.
(2) Secondly, assume that n has odd number of divisors; we should prove that it is a full square.
In fact, we will prove the counterpositive: If n is **not** a full square, then it has even number of divisors. To see this, we also arrange divisors of n in pairs – as before, one of them is less than $\sqrt{n}$, another one is bigger than $\sqrt{n}$. Only this time the number $\sqrt{n}$ itself does not count as a divisor of n, because it is not an integer.
Example: The number $12 \approx (3.4641\ldots)^2$ has these divisor pairs:

$$(1;12),\ (2;6),\ (3;4).$$