Tutorial 8

assertions

Q1

```
assertion_1: assert property(@ (posedge clk) not(a && b) );
```

Q2

```
assert property (@(posedge clk) req |-> ##3 ack);
```

Q3

```
assertion_2: assert property(@(posedge clk) a ##1 b[*3] |->
   DO == DI &&
   $past(DO,1) == $past(DI,1) &&
   $past(DO,2) == $past(DI,2));
```

Q4

```
assert property ((@posedge clk) vld |-> (cnt == ($past(cnt) + 1));
```

Q5

```
prpoerty p;
  @(posedge clk) b|-> ($past(a,2) == 1);
endproperty

assertion-3: assert property(p)
```

Q6

```
property p
  (@posedge clk)
  $rose(a) && state == ACTIVE ##1
  !b[*1:$] ##1
  $rose(a) && state == ACTIVE |->
  ##[1:3]state == START;
endproperty

assertion_4: assert property (p);
```

Q7

```
assert property (@(posedge clk) a|-> ##[1:$] b || c )
```

Q8

```
assert
  property(
    @(posedge clk) disable iff (rst) // sampling event
    req |-> ##3 gnt                  // expression to check
  )
  else                               // (optional) error message
    $error("%m no grant after request");
```

Q9

```
property Test;
  @(posedge clk) disable iff (reset)
  a |-> ##1 b[->3] ##1 c;
endproperty

newassert: assert property(Test);
```

Q10

```
assert property (@(posedge clk) disable iff reset
                 !req ##1 req[*1:$] ##1 !req
                 |->
                 !ack[#1:$] ##1 ack[*1:$]##1 !ack);
```