Assignment-01 Reporting Macros

1. Hello World Using UVM_INFO

Assume that you have a UVM testbench environment.

Tasks:

1. Create a UVM test named hello_test
2. Use uvm_info to display **"Hello World from UVM"**
3. Print message at **time = 0 ns**
4. Use **UVM_MEDIUM verbosity**
5. Stop simulation at **100 ns**

2. Timed Hello World Messages

Assume a UVM test is running.

Tasks:

1. Print "Hello World – Start" at 0 ns
2. Print "Hello World – 50 ns Reached" at 50 ns
3. Use UVM_LOW verbosity
4. Generate a warning at 75 ns
5. Stop simulation at 150 ns

3. Hello World with Severity Levels

Tasks:

1. Print "Hello World – INFO" using uvm_info at 10 ns
2. Print "Hello World – WARNING" using uvm_warning at 40 ns
3. Print "Hello World – ERROR" using uvm_error at 80 ns
4. Observe simulator behavior
5. Stop simulation at 120 ns

4. Variable Initialization and Hex Display

Assume that you have four variables initialized to the constant values as mentioned below:

1. reg [3:0] a = 4'b1010
2. reg [15:0] b = 16'h1122
3. integer c = 12
4. reg d = 1'b1

Tasks:

1. Use uvm_info to display all variable values
2. Display values in **hexadecimal format**
3. Print messages at **time = 0 ns**
4. Generate a uvm_warning at **50 ns**
5. Generate a uvm_error at **100 ns**
6. Forcefully stop the simulation at **200 ns** from testbench top

5. Timed Reporting with UVM Severities

Assume a UVM testbench environment.

Tasks:

1. Display **"Simulation Started"** using uvm_info
2. Use **UVM_MEDIUM verbosity**
3. Generate a warning at **40 ns**
4. Generate an error at **80 ns**
5. Stop the simulation at **150 ns**

6. Loop-Based UVM Reporting

Assume an integer variable count.

Tasks:

1. Use a for loop from `count = 0` to `count = 9`
2. Print each value using `uvm_info`
3. Generate `uvm_warning` when `count == 5`
4. Generate `uvm_error` when `count == 8`
5. Stop the simulation after loop execution

7. Multiple Severity Scheduling

Tasks:

1. At **30 ns** → print `uvm_info`
2. At **60 ns** → generate `uvm_warning`
3. At **90 ns** → generate `uvm_error`
4. At **120 ns** → call `$fatal`
5. Observe severity handling in simulator