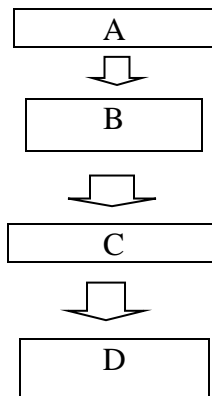


System Verilog Assignment Questions

ASSIGNMENT PROGRAMS

ASSIGNMENT-1

1. Write a program to enter even numbers with array size 30 elements using functions inside a class and display the result.
2. Write a program to enter same value into array size 10 elements using functions inside a class and display the result.
3. Write a program
 - a. Take 2 arrays of different size.
 - b. Enter the values to both the arrays using single function
 - c. Compare both the array values at each index
 - d. Display the values present at index of both array.
 - e. Display the number of elements matched in the array [count].
4. Design a ALU inside a class.
 - a. Write program to perform operations [+*/] and display result using Scope Resolution Operator.
5. Write a Program and Display
 - a. Take 2 Queues in a class
 - b. Enter the values into 2 Queues using “THIS” operator.
 - c. Print both Queues.
 - d. Do Push front and pushback value operation using FUNCTION and THIS operator.
6. Write a Program for Hierarchical class .
 - a. Take 2 Properties and Methods in each class.
 - b. Structure



- c. Access all the methods and properties using class D handle.
7. Do SHALLOW COPY and DEEP COPY
 - a. Take 3 parent class and 1 child class.
 - b. Create 3 handles for child class and check the values of properties of parent class when they are modified using handles.
8. Write a program consists of

- a. One string static variable and int static variable and static method.
 - b. Perform operation of int variable inside static method and display the result of static string and string properties inside method using scope resolution.
9. Inheritance:
 1. Take a scenario of some properties in 1 parent class ,2 child class.
 2. Take new function in both parent and child.
 3. Try to access new function of parent inside function of both childs.
- 10 . Polymorphism:

Take a Parent class with 2 functions

 1. Mobile Specification 2. Print specifications.
 - a. Take 3 child classes of different mobile company.
 - b. Print Specs of respective child[mobile company].

ASSIGNMENT –2

1. Write a Program that includes Static variables and Static Methods?
2. Write a Program for the concept of Polymorphism with and without Virtual Keywords and Handle declarations?
3. Explain the concept of SV Environment for UART?
4. Write Syntax for All the types of Constraints?

ASSIGNMENT – 3

1. Write a self-checking test bench for any DUT, using if-else statement and \$display to print PASS or FAIL cases.
2. Create a report on the outputs of the above question using file operations.

ASSIGNMENT – 4

1. Write a system verilog program to print 2 strings in concatenated fashion.
2. Write a SV program on ‘enum’ having 6 colors – red, pink, black, blue, yellow, green. Print all enum member names along with its values. [Hint: use for loop]
3. Repeat the above question with initialized enum.
4. Compare 2 instances of enum created in the 2 & 3 question and print result.

ASSIGNMENT – 5

1. Create a dynamic array, allocate 5 locations, fill the array and print it. Again allocate 10 locations to same array and check whether the location is overwritten or not, by printing the values in it.
2. Write a SV code to check the methods (new, size, delete) of dynamic array.
3. Take a fixed size array – ‘{4,8,3,5,2,9}’ and use any logic to sort then elements.
4. Take an array and initialize it. Copy this array into another array and compare it. Also check by slicing the elements.

ASSIGNMENT – 6

1. Print a 4-bit grey code as 16-cell k-map value in 4x4 pattern.
2. Create a 32x32 bit memory using dynamic array and fill memory with your own data & read them. Then extend it to 32x64 memory.
3. Write a SV program using associative array of 8 elements of integer type, display the result and size.
4. Take an arr[8]='{5,6,8,3,4,9,7,2}' as input. Sort and store in queue 'q1[\$]' and display.

ASSIGNMENT – 7

1. Write a SV code using final block and check it.
2. Write a SV code using priority-if and unique-if statements, and check it.
3. Write a SV code with jump statements and check it.
4. Write any DUT in verilog and its test-bench in SV, and check it.

ASSIGNMENT - 8

1. Check for Named block and disable statement.

ASSIGNMENT – 9

1. Check the code for all the process and process control statements, written in the following format:
(NOTE: fork-join, fork-join_all, fork-join_none, disable fork, wait fork)

Initial begin	process block --
Stmt 1; // display statements	p1: begin
Stmt 2;	#10; \$display ("stmt");
Fork	#5; \$display ("stmt");
Process1;	end: p1
Process2;	
Process3;	similarly p2 and p3 block.
Join	
Stmt3;	
Stmt4;	
end	

ASSIGNMENT – 10

1. Write a code using function and call it inside fork-join block, and check it.
2. Write a code for all types of arguments passing.

ASSIGNMENT – 11

1. Write a code using class assignment and check whether pkt1, pkt2, pkt3 are getting affected when any one of them changes.

ASSIGNMENT – 12

1. Check by writing a code for Super keyword.
2. Check by writing a code for Shallow copy and Deep copy.
3. Check by writing a code for data hiding – local and protected keywords.

ASSIGNMENT – 13

1. Check a code for randomization with rand and randc.
2. Write a TB for 4:2 encoder (DUT) using randomization.
3. Check the above DUT by applying 4 patterns – 1000, 0100, 0010, 0001. (Hint: use if-else stmt)
4. Check the above code with disable rand.

ASSIGNMENT – 14

1. Write a code for internal constraint for your desired range of address and display the output.
2. Write code for external constraint with 2 different modules and display the output.[HINT: write class in one file and create 2 modules named file1 and file2, and call the class using `include`]
3. Write a code keeping encoder as DUT, print output for required patterns using inside operator.
4. Write a code for distribution constraint and check the output.

ASSIGNMENT – 15

1. For encoder DUT, write a code using constraint implication and generate addr_range with if-else statement for valid inputs only.
2. Check the above code by disabling constraint after 8 times and print randomized values.
3. Execute a code with and without soft constraint and analyze the difference.
4. Design a simple 4-bit adder DUT and create a test environment such that the output generated doesn't exceed 15 using bidirectional constraint.

ASSIGNMENT – 16

1. Execute semaphore code with \$display statements included in between process statements to check the order of statements execution.
2. Check the above code without put() method.
3. Execute a semaphore code containing 3 processes with 4 keys, but available only 2 keys to processes.
4. Execute a semaphore code with only 1 key controlling all the 4 processes.
5. Execute a semaphore code containing 4 keys with 2 processes using try_get() method.
6. Write a code using 1 semaphore variable and check whether the data between 4 processes is overridden or not.
7. Create 2 functions such that one function will put data in mailbox and the other gets data from mailbox.
8. Create 2 processes and check event getting triggered or not.

ASSIGNMENT – 17

1. Create an interface without using modport.

2. Use modport to create a testbench for counter with clocking block and with randomized inputs.

ASSIGNMENT – 18

1. Design an encoder as DUT, taking default value as $y = 2'bzz$ and use assert & \$display for it.
2. For the above question, use assert with condition $y=2'bzz$ and \$display followed by else condition and \$display.

ASSIGNMENT – 19

1. Check all the example codes for implication operator(both overlapping and non-overlapping), consecutive operator and go-to repetition operator.
2. Write code for code for checking gates using \$past taking a simple block containing input a, clk and output y.

ASSIGNMENT – 20

1. Check for explicit coverage and automatic bins.
2. Check for bins for transition by creating separate class for each bins mentioned in example.