1) Using gate level implementation, Design the module `Full_Adder`, in *Verilog*. Then, based on this module construct the following adders:

a. 4-bit Ripple Adder

b. 4-bit Carry Look Ahead Adder

c. 16-bit Ripple Adder

d. 16-bit Carry Look Ahead Adder

Write a *testbench* for each of the 16-bit adders above and compare their speeds. Use the gate propagation delays given in the table below:

Gate Type	Normalized Delay
AND	1.4
OR	0.9
XOR	1.3
NOR	1.5
NAND	1.1

[Hint: Make use of `for` loops or `while` loops or `generate` to replicate hardware logic.]

2 – Bonus point) Design a module, `Fibonacci_Sequence`, in *Verilog* that takes two inputs, `Clock` and `reset`, and outputs the Fibonacci sequence. The output must follow the pattern of Fibonacci (i.e. The sequence must start with two 1s).

Write a *testbench* for your module and examine the generated waveforms.

Notes:

Send all your assignment related files (*Top module* and *Testbench* [.v] and [.vvp] and [.vcd] files) along with a detailed report in [.pdf] format all in one zip file to the email address of the class.

For this assignment you can work in groups of two, each participant must submit the assignment individually with their respective name and student number.

If you have any questions regarding this assignment, feel free to contact us.

Please submit your homework, simulations and projects in the following format:

Name_StudentNumber_Verilog_HW0 (BillGates_12345678_Verilog_HW0)

Good Luck!

Spring 2023