**DAILY ASSESSMENT FORMAT**

|  |  |  |  |
| --- | --- | --- | --- |
| **Date:** | **29/05/2020** | **Name:** | **Yalpi Nandika** |
| **Course:** | **Logic Design** | **USN:** | **4AL17EC096** |
| **Topic:** | **Analysis of clocked sequential circuits.**  **Digital clock design** | **Semester & Section:** | **6th sem**  **B section** |
| **GitHub Repository** | **Yalpi-Online-Courses** |  |  |

|  |
| --- |
| **FORENOON SESSION DETAILS** |
| **Image of session** |
| **\*analysis of clocked sequential circuits**  **The behavior of a clocked sequential circuit is determined from its inputs, outputs and state of the flip-flops (i.e., the output of the flip-flops). The analysis of a clocked sequential circuit consists of obtaining a table of a diagram of the time sequences of inputs, outputs and states.**  **The clocked sequential circuits have flip-flops or gated latches for its memory elements. There is a periodic clock connected to the clock inputs of all the memory elements of the circuit to synchronize all the internal changes of state.**  **\*digital clock design**  **In a digital clock, this is usually provided by using a crystal which is made out of glass. As an electric charge passes through the crystal, it will change shape slightly and make a very light sound. The sound which is heard at a regular frequency is then converted into an electronic signal.**  **An analog watch might be much more accurate than a digital one if it uses a high-precision movement to measure passing time. Generally, the most expensive watches in the world are analog ones, though the world's most accurate atomic clocks show time with digital displays** |

|  |  |  |  |
| --- | --- | --- | --- |
| **Date:** | **29/05/2020** | **Name:** | **Yalpi Nandika** |
| **Course:** | **Python** | **USN:** | **4AL17EC096** |
| **Topic:** | **Generators with python** | **Semester & Section:** | **6th sem**  **B section** |
| **AFTERNOON SESSION DETAILS** | | | |
| **Image of session**      **\*generation with python**  **Python generators are a simple way of creating iterators. All the work we mentioned above are automatically handled by generators in Python. Simply speaking, a generator is a function that returns an object (iterator) which we can iterate over (one value at a time).**  **What is a Python Generator (Textbook Definition) A Python generator is a function which returns a generator iterator (just an object we can iterate over) by calling yield . yield may be called with a value, in which case that value is treated as the "generated" value.**  **Generators have been an important part of Python ever since they were introduced with PEP 255. Generator functions allow you to declare a function that behaves like an iterator. They allow programmers to make an iterator in a fast, easy, and clean way. ... An iterator is an object that can be iterated (looped) upon.** | | | |

**Webinar:**

**Topic: Preparing for the Next Normal**

**Image of session:**







