# An-Najah National University
# Faculty of Engineering and Information Technology
# Computer Engineering Department

## *Digital Circuit Design II (10636321)*
## *Due to 22/11/2021*

| ***HW1***: (ILOs: **III**) | Points: **20** |
|---|---|

Complete the following VHDL code to implement a system that searches for a specific pattern of bits (X) within another given data (***D***). Your system should reverse the bit sequence in each occurrence of the given pattern. The result (Q) should include the give data after modification.

Your system should fulfill the following requirements:

- If the enable input is asserted to '1', the system will start the search and replace processes on the rising edge of the input clock.
- The Asynchronous reset has the highest priority. When it is activated (reset =1), the outputs are cleared.
- 

**e.g. X= 1010 D** = 1**1010**11100**1010**000101 ➔ **Q** = 1**0101**11100**0101**000101

```
entity SearchAndReverse is
  port(     CLK : in  std_logic;          -- clock input
            Reset: in  std_logic;         -- Active High Asynchronous Reset
            Enable  : in  std_logic;      -- synchronous enable
            D   : in  std_logic_vector(19 downto 0); -- 20 bits input data
            X   : in  std_logic_vector(3 downto 0);  -- 4 bits pattern
            Q   : out std_logic_vector(19 downto 0)  --20 bits output
            );
end SearchAndReverse;
```

**You have to submit two files**:

1. A VHDL code to implement your design (**14 Points**)
2. A testbench file to simulate and test your design. (**6 Points**)

You should cover the following test cases:

| Reset | D | X | Enable | Duration |
|---|---|---|---|---|
| 1 | - | - | - | 3 clock cycles |
| 0 | - | - | 0 | 3 clock cycles |
| 0 | 10101010101010101010 | 1010 | 1 | 3 clock cycles |
| 0 | 10101010101010101010 | 0101 | 0 | 3 clock cycles |
| 0 | 11111111111111111111 | 0000 | 1 | 3 clock cycles |
| 0 | 00001111000011110001 | 0011 | 1 | 3 clock cycles |