MECHENG 709/710 – Assignment Question Sheet (2025)

The main aim of this part of the assignment is to help you understand some of the basic elements in a set of OPC UA Python codes, including address space and namespace that make up an important part of an OPC UA information model.

This part of the assignment is worth **8 marks** in total.

# Practice questions (not marked)

To answer the questions in this part, you have to study, understand and run the given example server and client codes in the “e*xample codes”* folder (*example\_server.py & example\_client.py).* They can be found on Canvas. Assume you initialise the server before starting the client.

1. What is the endpoint URL for the server, and what is the endpoint URL for the client?
2. What is the root node ID for the server?
3. What are the browse names, default values, and node IDs for the following two assigned variables?

|  |  |  |  |
| --- | --- | --- | --- |
| **Python Variables** | **Browse name** | **Default value** | **Node ID** |
| Sensor\_name |  |  |  |
| Temperature |  |  |  |

# Questions related to the assignment codes (8 marks)

These questions are based on the following codes:

* *OPC\_UA\_Server.py*
* *Company1\_Client.py*
* *Company2\_Client.py*

1. In the ***python-opcua*** library, many useful *Methods* (class functions) are defined. These methods allow developers to create OPC UA clients and servers, perform read and write operations, subscribe to data changes, and more. Please fill in the *Method Call* in the table below based on the description of common methods (the first row provides an example). (*2 marks*)

|  |  |  |
| --- | --- | --- |
| **No.** | **Description of common methods** | ***Method Call*** |
| 1 | Assign an endpoint URL to the OPC UA Server | *set\_endpoint()* |
| 2 | Get the browse name of a node in the OPC UA Server |  |
| 3 | Get the value of a node in the OPC UA Server |  |
| 4 | Create a variable for an object in the Server |  |
| 5 | Invoke a method on the OPC UA server (used by an OPC UA client) |  |

1. The following are partial node details extracted from the OPC UA Client *Company1\_Client.py*. This includes the name, browse name, and node ID under the variables. Please fill in the remaining blanks in the table based on the example in the first row. No additional rows are required. (*4 marks*)

If needed, consult *[Lecture notes and References Book of OPC Unified Architecture]*

|  |  |  |  |
| --- | --- | --- | --- |
| **No.** | **Variable Name**  **(Company1 Client)** | **Browse Name** | **Node ID** |
| 1 | Equipment\_ID1 | Equipment\_ID | ns=2;i=2 |
| 2 | Equipment\_ID3 |  |  |
| 3 | Lathe\_operation |  |  |
| 4 | WorkpieceID |  |  |
| 5 | Kuka\_Status |  |  |
| 6 |  | remaining\_con |  |
| 7 |  | remaining\_Kuka |  |
| 8 |  | remaining\_Lathe |  |
| 9 |  | Time\_Stamp |  |

1. Fill the vacant cells in the table below based on the codes *Company1\_Client.py & Company2\_Client.py*. Below is a collation of the callable method names, browse names and NodeID extracted from the OPC UA server under the Node Class object. (If some details are not applicable, enter **N/A**) (*2 marks*)

If needed, consult *[Lecture Notes and References Book of OPC Unified Architecture]*

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **No.** | **Callable Methods (both clients)** | **Browse Name** | **Node ID** | **Function Name** | **Argument** |
| **1** | Start\_Conveyor\_prog | Conveyor | ns=1;i=2001 | Start\_Conveyor\_prog | Current\_operation |
| **2** |  |  |  |  |  |
| **3** |  |  |  |  |  |
| **4** |  |  |  |  |  |
| **5** |  |  |  |  |  |