

## Assignment 1 - CS661A

### VTK data loading, Cell data processing, and Visualization

Due date: Jan 25, 2023, 11:59pm

Grade: 50 points (5% of the course grade)

In this assignment, you will learn how to load a 2D uniform grid data stored as `VtkImageData`. You will learn how to query various types of information about the dataset, such as the dimension, the number of points, number of cells, etc., and finally you will extract one cell from the data and create a new `VtkPolyData` object and visualize the vertices of the extracted cell using VTK's rendering capabilities.

#### Dataset:

The dataset that you will use in this assignment is a 2D slice taken from a 3D Scalar field of a Hurricane Simulation Data. The variable is Pressure. If you want to know more about the original data, please refer to this link: <http://vis.computer.org/vis2004contest/index.html>

#### Here are the tasks for this assignment:

Load the dataset using `vtkXMLImageDataReader()` and create a data object from the reader's output. Note that the dataset contains a scalar attribute array named 'Pressure,' i.e., each point in the data set has a Pressure value, and the values are stored in a `VtkDoubleArray` in the dataset.

#### 1. Data Query/Processing Task: [30 Points]

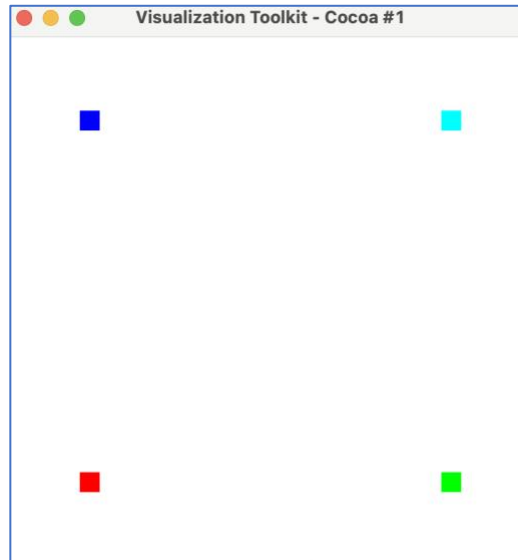
Print the following information about the dataset:

- Number of cells in the dataset
- The dimensions of the dataset
- The number of points present in the uniform grid of the data. (Think about the relationship between the total number of points and the data dimension)
- Print the range of Pressure values present in the dataset.
- Print the average Pressure value of the entire dataset, i.e., the Pressure array. For this, you will have to access the Pressure values and compute the average of all points.
- Extract a `vtkCell` object with cell id=0, i.e., the first cell. This is a quad cell, so the cell will have four vertices and four edges, and one face. Note that, the cell id in your program should be a parameter and if the user changes the cell id, then all the quantities that you are computing about the cell should get updated.
- Print the indices of the four corner vertices of the cell.
- Print the 3D coordinate of each vertex. Note that this is a 2D dataset, but all the points are stored as 3D points in the VTK data file with a constant Z-coordinate value=25.
- Compute the 3D coordinate of the cell center using its four corner vertices. The center location can be computed as the average of the corner vertices. Again, note that the Z coordinate of the center will be 25.
- Print the data/attribute value (Pressure) for all the four vertices of the extracted cell.
- Compute and print the mean (average) Pressure value at the cell center by averaging Pressure values from the four cell vertices.

#### 2. Visualization Task: [20 Points]

Assuming you can access the information for the above task, you already know the coordinates of the four corner vertices of the cell that you have extracted with cell id=0. The coordinates are 3D points with a constant Z coordinate value=25. Now create a new `VtkPolyData` object, add these 4 points into the `VtkPolyData` and specify a separate color for each point. For example, you can represent the red

color as (255,0,0) as an RGB 3-value tuple. For help assign a color to the points, you can consult the ColoredTriangle example [1] as a reference. Next, use `vtkVertexGlyphFilter()` to create visual representations for each point in the form of a Vertex Glyph. Finally, use VTK mapper, actor, and renderer to show the output of the `vtkVertexGlyphFilter()` as points in the screen. If you are doing it right, your output should look like the following image. Note that you may have different colors for vertices depending on what color you have assigned to them in your code.



### References:

[1] <https://kitware.github.io/vtk-examples/site/Python/PolyData/ColoredTriangle/>

[2] <https://vtk.org/doc/nightly/html/classvtkImageData.html#details>: You can use the documentation of the `VtkImageData` class from this link to find out the methods that are defined for this class that you can use in your code.

[3] Consult the sample VTK codes provided to you. You can already find several hints from those codes.

**\*\* Please note that the successful completion of this assignment will prepare you for the next assignment. You do not need any other Python library to solve this assignment. Please spend some time and get familiar with the VTK APIs so that you can manipulate and query data properties and develop an understanding of how a VTK code is typically structured and how VTK data sets are handled.**

### How to submit?

The **HelloIITK** portal will be set up for submission. There will be a time limit set, and if you miss the deadline, you will start losing points as per the late submission policy. If you miss the deadline, please contact me, and I will deal with it on a case-by-case basis. Please start early and try to finish it by the deadline. You should submit your code as Python scripts or a Jupyter Notebooks with any other relevant files into a single compressed (\*.zip) file. Your code should be commented properly, and when printing any information as part of any assigned task, please mention the information that is being printed. Compress your solution files and name your submission as “**Lastname\_rollnum\_Assignment1.zip**”.