BLOCK DIAGRAM FOR THE PROCESS:

Step 1: Obtain the Factual Explanation Subgraphs from all instances belonging to a particular Class using a Factual Graph Explainer (Ex: CFF, RC Explainer):

- The embedding space of the graph instances as well as the subgraph explanations will depend on the GNN architecture used for the prediction in that framework, so the explanation subgraphs produced by different frameworks may vary.
- We aim for a generalised approach to produce global counterfactual explanations using factual explanation subgraphs produced using some Factual Explainer -> so our approach (well ideally should be) agnostic to the Factual Explainer used to produce the factual instance-specific explanations.

Step 2: Obtain the Induced Subgraph Explanations from all instances belonging to a particular Class, create their vector embedding and cluster them:

- Try the 2 approaches for explanation subgraph extraction:
  - **First (More preferable) :** Extract each disconnected explanation subgraph within a graph instance as a separate explanation subgraph -> so each instance may have multiple disconnected explanation subgraphs (as generated by the Explainer) -> and each such sugraph will be embedded seperately before clustering
    Reason for higher preference: Extracts all possible prototypes and may lead to better seperable clusters
  - **Second :** Take all the explanation subgraphs (even if they are disconnected) as one subgraph explanation for a particular instance -> so overall there is only 1 explanation per instance of a class -> thus there will be one embedding containing the info of factual explanation of one graph instance
- Use the same GNN model as used by the Factual Explainer to create the embedding of explanation subgraphs before clustering (For Ex: GCN in CFF Explainer)
- Use a **hierarchial clustering approach** that determines the **no of clusters** from the data points itself (rather than hardcoding the no of clusters) : **For Example : X-means Clustering, Elbow method, Silhoutte coefficient)**
- **Finding cluster representative post clustering:** For each cluster , find the mean embedding of all the subgraphs of that cluster, and the subgraph closest to the mean embedding based on some distance (L2/L1) metric is the representaive global factual explanation for that cluster.

Step 3: Using the info from the clustering of both class 0 and class 1 to produce global counterfactual explanations:

Two parts: First, generating CF explanations for class 0, then same process to produce CF explanations for class 1:

Producing CF candidates for class 0 first:

- For each cluster in class 0, after having identified its representative subgraph: Iteratively add/remove edges/nodes (basically iteratively do a node/edge perturbation) from the representative.
  Criterion for edge/node perturbation at each step: At each step, choose the edge/node perturbation which maximizes a graph similarity measure (say SED or an any neural approximation of it) to the mean embedding of the class –1 clusters (i.e. take all the class 1 cluster representatives and take the mean of their embeddings -> to get class 1 mean embedding).
- Also after each such perturbation, check if the prediction for the current class (class 0) decreases as it each iteration , and as soon as the prediction flips, stop and don't do further perturbation. This is the global CF explanation for that cluster (and so for the graphs whose atleast one explanation subgraph was in that cluster)
- Next, for each original graph instance that had a subgraph in this cluster -> we need to identify that representative subgraph within that cluster , make the same perturbations to this subgraph within that graph instance and check if the prediction flips for this entire instance -> we need to check for how many of the original instances tied to this cluster, the prediction flips this way: if it happens for a majority of them -> then the global CF is a valid one and this method somewhat works.
- Repeat the above process for each cluster in class 0.

Follow the same procedure now with class 1 clusters to generate CFs for class 1 instances: