WEEK 15

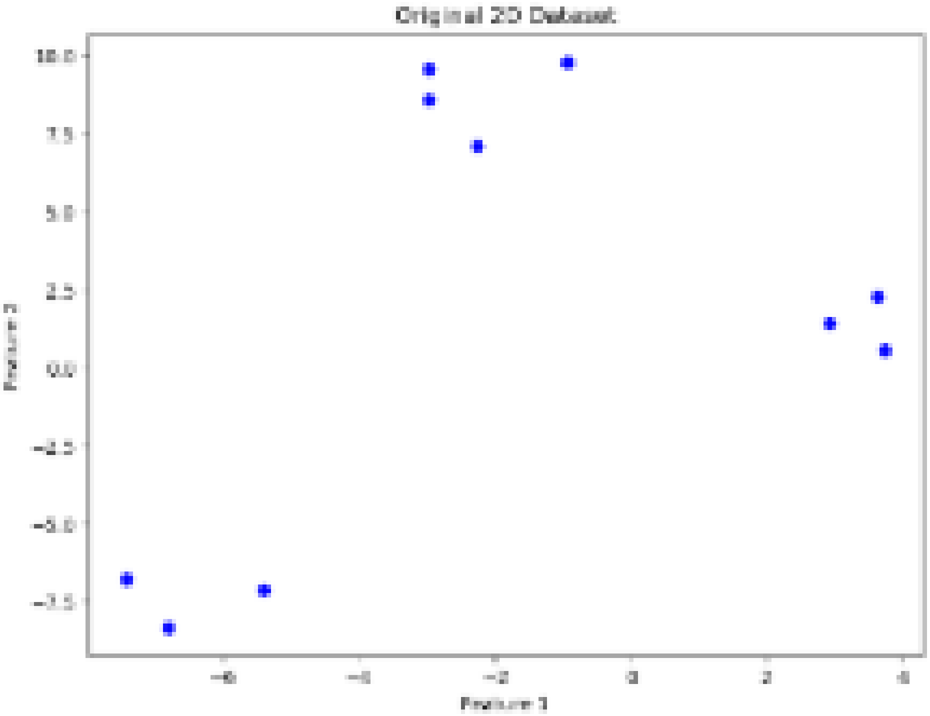
71.Write a program that performs Agglomerative Clustering on a 2D dataset and outputs the number of clusters formed at each iteration of the merging process.

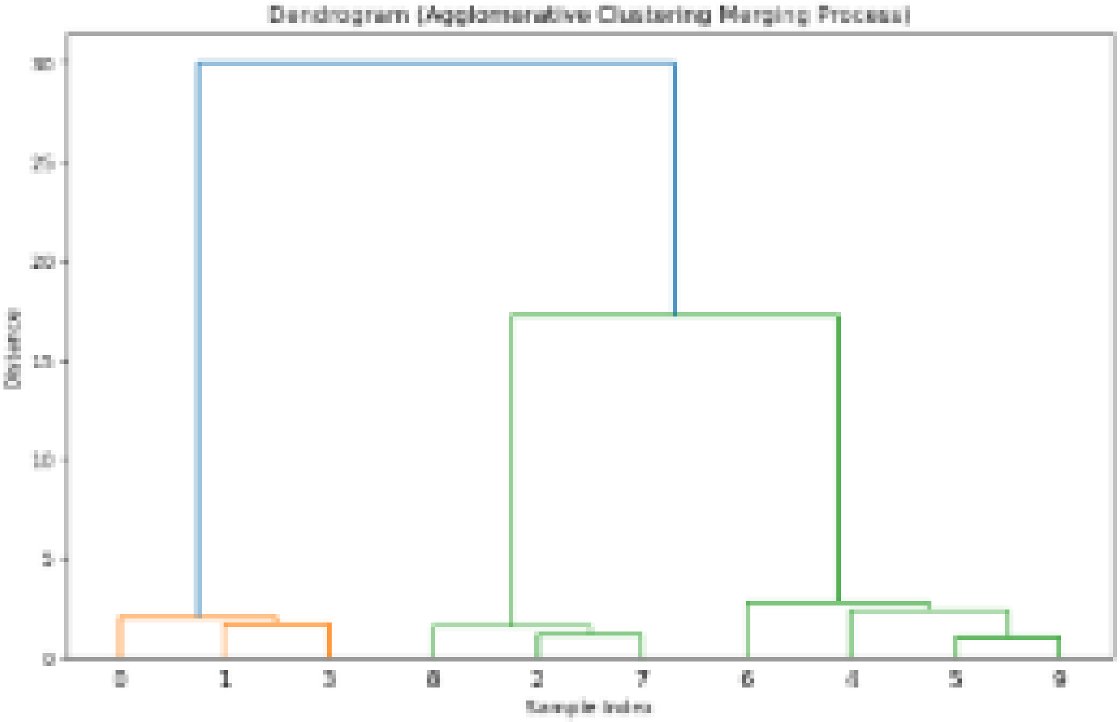
```
import numpy as np
import matplotlib.pyplot as plt
from scipy.cluster.hierarchy import linkage, fcluster

# Step 1: Generate a 2D dataset (using make_blobs to create 2D clusters)
from sklearn.datasets import make_blobs
X, _ = make_blobs(n_samples=10, centers=3, random_state=42)

# Step 2: Perform Agglomerative Clustering using linkage function
Z = linkage(X, method='ward')  # 'ward' minimizes the variance of merged clusters

# Step 3: Track the number of clusters formed at each iteration
# Initially, there are as many clusters as data points
num_clusters = len(X)

# Step 4: Print the number of clusters at each iteration
print("Number of clusters at each iteration of merging:")

# Iterating through the linkage matrix Z
for i in range(len(Z)):
    num_clusters -= 1  # After each merge, the number of clusters decreases by 1
    print(f"Iteration {i + 1}: Number of clusters = {num_clusters}")
```

```
# Step 5: Visualize the original dataset
plt.figure(figsize=(8, 6))
plt.scatter(X[:, 0], X[:, 1], c='blue', marker='o')
plt.title("Original 2D Dataset")
plt.xlabel("Feature 1")
plt.ylabel("Feature 2")
plt.show()

# Step 6: Dendrogram plot for visualizing the merging process
from scipy.cluster.hierarchy import dendrogram

plt.figure(figsize=(10, 6))
dendrogram(Z)
plt.title("Dendrogram (Agglomerative Clustering Merging Process)")
plt.xlabel("Sample Index")
plt.ylabel("Distance")
plt.show()
```

Output

```
Number of clusters at each iteration of merging:
Iteration 1: Number of clusters = 9
Iteration 2: Number of clusters = 8
Iteration 3: Number of clusters = 7
Iteration 4: Number of clusters = 6
Iteration 5: Number of clusters = 5
Iteration 6: Number of clusters = 4
Iteration 7: Number of clusters = 3
Iteration 8: Number of clusters = 2
Iteration 9: Number of clusters = 1
```





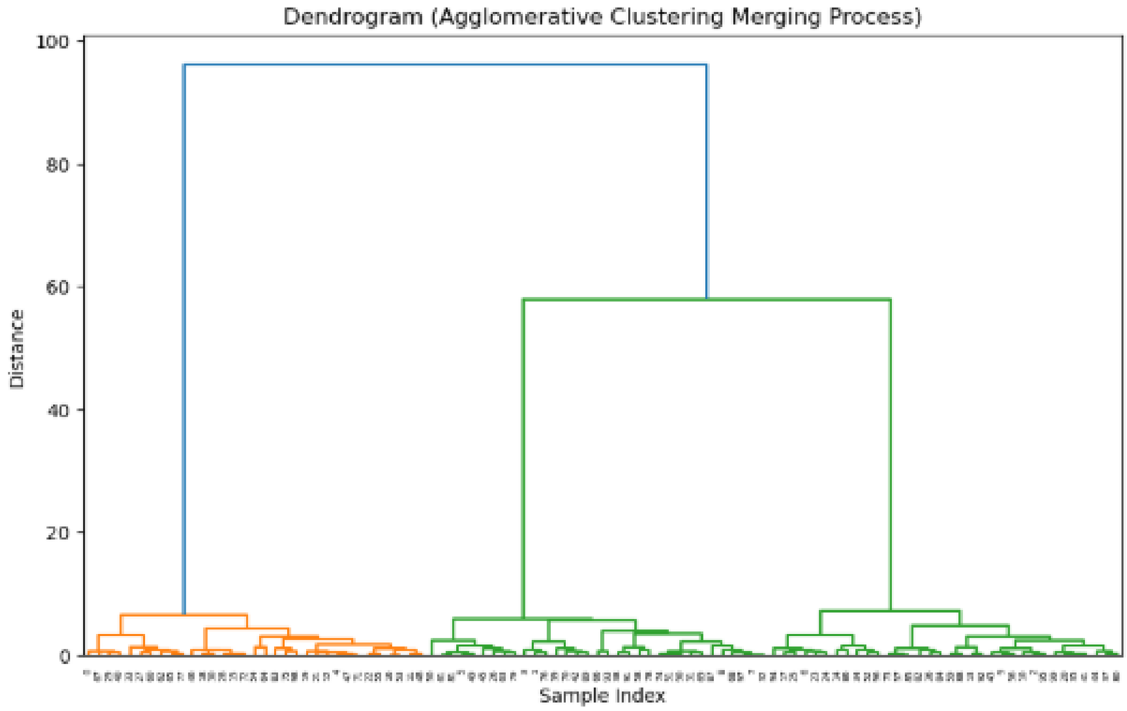
72.Write a program that takes a dataset of 2D points and performs Agglomerative Clustering. The program should allow the user to visualize the merging process and the final clusters.

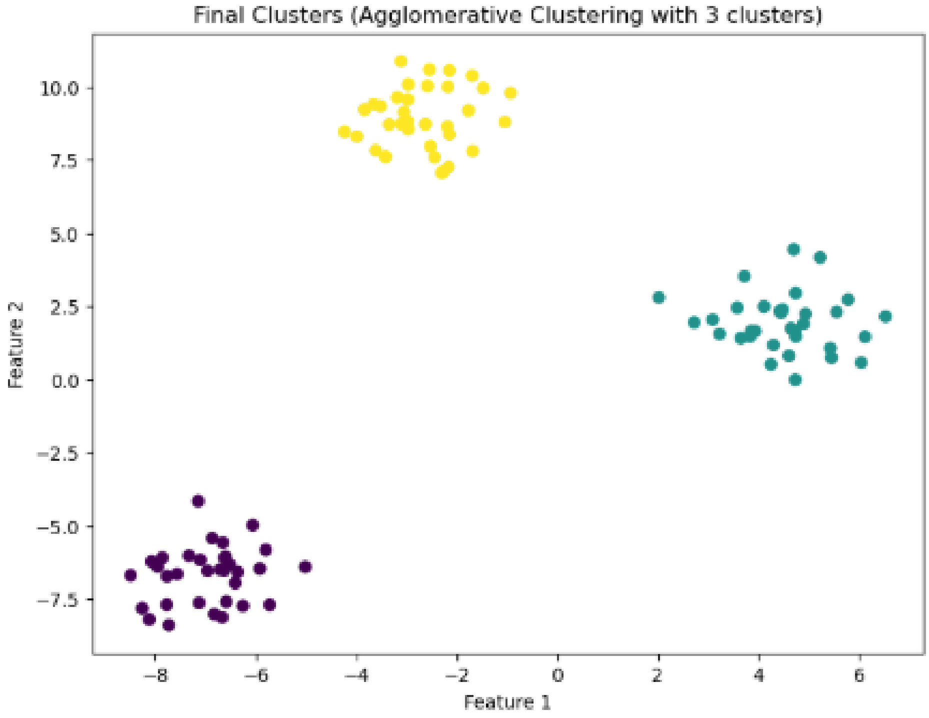
```
import numpy as np
import matplotlib.pyplot as plt
from sklearn.datasets import make_blobs
from scipy.cluster.hierarchy import linkage, fcluster, dendrogram

# Step 1: Generate a 2D dataset
def generate_data():
    # Generate synthetic data (you can also replace this with your dataset)
    X, y = make_blobs(n_samples=100, centers=3, random_state=42)
    return X

# Step 2: Perform Agglomerative Clustering
def perform_agglomerative_clustering(X, num_clusters=3):
    # Perform hierarchical/agglomerative clustering
    Z = linkage(X, method='ward')  # 'ward' minimizes variance of clusters
    return Z

# Step 3: Visualize the Dendrogram
def plot_dendrogram(Z):
    plt.figure(figsize=(10, 6))
    dendrogram(Z)
    plt.title("Dendrogram (Agglomerative Clustering Merging Process)")
    plt.xlabel("Sample Index")
    plt.ylabel("Distance")
    plt.show()
```

```
# Step 4: Assign Final Clusters based on desired number of clusters
def plot_final_clusters(X, Z, num_clusters):
    # Cut the dendrogram at the desired number of clusters
    clusters = fcluster(Z, t=num_clusters, criterion='maxclust')

    # Plot the data points colored by cluster
    plt.figure(figsize=(8, 6))
    plt.scatter(X[:, 0], X[:, 1], c=clusters, cmap='viridis', marker='o')
    plt.title(f"Final Clusters (Agglomerative Clustering with {num_clusters} clusters)")
    plt.xlabel("Feature 1")
    plt.ylabel("Feature 2")
    plt.show()

# Main program to run Agglomerative Clustering
def main():
    # Generate a 2D dataset
    X = generate_data()

    # Perform Agglomerative Clustering
    Z = perform_agglomerative_clustering(X, num_clusters=3)

    # Visualize the Dendrogram
    plot_dendrogram(Z)

    # Visualize the final clusters
    plot_final_clusters(X, Z, num_clusters=3)
```

```
if __name__ == '__main__':

    main()
```

Output


Dendrogram (Agglomerative Clustering Merging Process)
Distance
Sample Index


Final Clusters (Agglomerative Clustering with 3 clusters)
Feature 2
Feature 1