Lab Program 7:

Apply **EM algorithm** to cluster a set of data stored in a .CSV file. Use the same data set for clustering using ***k*-Means algorithm**. Compare the results of these two algorithms and comment on the quality of clustering. You can add Java/Python ML library classes/API in the program.

K-means:

```
from sklearn import datasets

from sklearn.metrics import accuracy_score

from sklearn.metrics import confusion_matrix

import matplotlib.pyplot as plt

import pandas as pd

import numpy as np


from sklearn.cluster import KMeans


iris = datasets.load_iris()


x = pd.DataFrame(iris.data)

x.columns = ['Sepal_Length','Sepal_Width','Petal_Length','Petal_Width']


y = pd.DataFrame(iris.target)

y.columns = ['Targets']


plt.figure(figsize=(14,7))

colormap = np.array(['red','lime','black'])

plt.subplot(1,2,1)

plt.scatter(x.Sepal_Length, x.Sepal_Width, c = colormap[y.Targets], s=40)
```

```
plt.title('Sepal')
plt.subplot(1,2,2)
plt.scatter(x.Petal_Length, x.Petal_Width, c = colormap[y.Targets], s=40)
plt.title('Petal')

model = KMeans(n_clusters = 3)
model.fit(x)

plt.figure(figsize=(14,7))
colormap = np.array(['red','lime','black'])
plt.subplot(1,2,1)
plt.scatter(x.Petal_Length, x.Petal_Width, c = colormap[y.Targets], s=40)
plt.title('Real')
plt.subplot(1,2,2)
plt.scatter(x.Petal_Length, x.Petal_Width, c = colormap[model.labels_], s=40)
plt.title('kmeans')

print("accuracy_score", accuracy_score(y.Targets, model.labels_))
print("confusion_matrix\n", confusion_matrix(y.Targets,model.labels_))
```

EM Program:

```
from sklearn import datasets
from sklearn.metrics import accuracy_score
from sklearn.metrics import confusion_matrix
import matplotlib.pyplot as plt
import pandas as pd
```

```
import numpy as np

from sklearn import preprocessing
from sklearn.mixture import GaussianMixture

iris = datasets.load_iris()

x = pd.DataFrame(iris.data)
x.columns = ['Sepal_Length','Sepal_Width','Petal_Length','Petal_Width']

y = pd.DataFrame(iris.target)
y.columns = ['Targets']

plt.figure(figsize=(14,7))
colormap = np.array(['red','lime','black'])
plt.subplot(1,2,1)
plt.scatter(x.Sepal_Length, x.Sepal_Width, c = colormap[y.Targets], s=40)
plt.title('Sepal')
plt.subplot(1,2,2)
plt.scatter(x.Petal_Length, x.Petal_Width, c = colormap[y.Targets], s=40)
plt.title('Petal')

scaler = preprocessing.StandardScaler()
scaler.fit(x)
xsa = scaler.transform(x)
xs = pd.DataFrame(xsa, columns = x.columns)
```

```
gmm = GaussianMixture(n_components = 3)
gmm.fit(xs)
y_gmm = gmm.predict(xs)

plt.figure(figsize=(14,7))
colormap = np.array(['red','lime','black'])
plt.subplot(1,2,1)
plt.scatter(x.Petal_Length, x.Petal_Width, c = colormap[y.Targets], s=40)
plt.title('Real')
plt.subplot(1,2,2)
plt.scatter(x.Petal_Length, x.Petal_Width, c = colormap[y_gmm], s=40)
plt.title('EM')

print("accuracy_score", accuracy_score(y.Targets, y_gmm))
print("confusion_matrix\n", confusion_matrix(y.Targets, y_gmm))
```