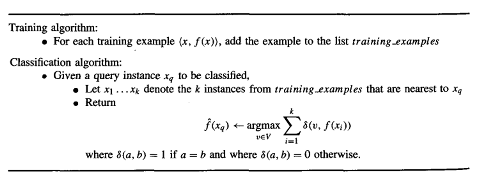
9.Problem : Write a program to implement k-Nearest Neighbour algorithm to classify the iris data set. Print both correct and wrong predictions. Java/Python ML library classes can be used for this problem.

Algorithm :





Program Code:

|  |
| --- |
| from sklearn.datasets import load\_iris  from sklearn.neighbors import KNeighborsClassifier  import numpy as np  from sklearn.model\_selection import train\_test\_split  iris\_dataset=load\_iris()  print("\n IRIS FEATURES \ TARGET NAMES: \n ", iris\_dataset.target\_names)  for i in range(len(iris\_dataset.target\_names)):  print("\n[{0}]:[{1}]".format(i,iris\_dataset.target\_names[i]))  print("\n IRIS DATA :\n",iris\_dataset["data"])    X\_train, X\_test, y\_train, y\_test = train\_test\_split(iris\_dataset["data"], iris\_dataset["target"], random\_state=0)    print("\n Target :\n",iris\_dataset["target"])  print("\n X TRAIN \n", X\_train)  print("\n X TEST \n", X\_test)  print("\n Y TRAIN \n", y\_train)  print("\n Y TEST \n", y\_test)  kn = KNeighborsClassifier(n\_neighbors=1)  kn.fit(X\_train, y\_train)    x\_new = np.array([[5, 2.9, 1, 0.2]])  print("\n XNEW \n",x\_new)    prediction = kn.predict(x\_new)    print("\n Predicted target value: {}\n".format(prediction))  print("\n Predicted feature name: {}\n".format  (iris\_dataset["target\_names"][prediction]))  i=1  x= X\_test[i]  x\_new = np.array([x])  print("\n XNEW \n",x\_new)  for i in range(len(X\_test)):  x = X\_test[i]  x\_new = np.array([x])  prediction = kn.predict(x\_new)  print("\n Actual : {0} {1}, Predicted :{2}{3}".format(y\_test[i],iris\_dataset["target\_names"][y\_test[i]],prediction,iris\_dataset["target\_names"][prediction]))  print("\n TEST SCORE[ACCURACY]: {:.2f}\n".format(kn.score(X\_test, y\_test))) |