Program 9

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.

Harsh R

In [18]:

```
import sklearn.datasets as dt
import pandas as pd
from sklearn.metrics import accuracy_score
```

In [19]:

```
iris=dt.load_iris()

X = pd.DataFrame(iris.data)
X.columns = ['sepal-length', 'sepal-width', 'petal-length', 'petal-width']
y = pd.DataFrame(iris.target)
y.columns = ['target']
```

In [ ]:

```
from sklearn.model_selection import train_test_split
from sklearn.neighbors import KNeighborsClassifier

xtrain, xtest, ytrain, ytest = train_test_split(X,y,test_size=0.2,random_state=4)

KRange = range(1,26)
scores = {}

for k in KRange:
    model = KNeighborsClassifier(n_neighbors=k)
    model.fit(xtrain, ytrain)
    ypred = model.predict(xtest)
    scores[k] = accuracy_score(ytest, ypred)
```

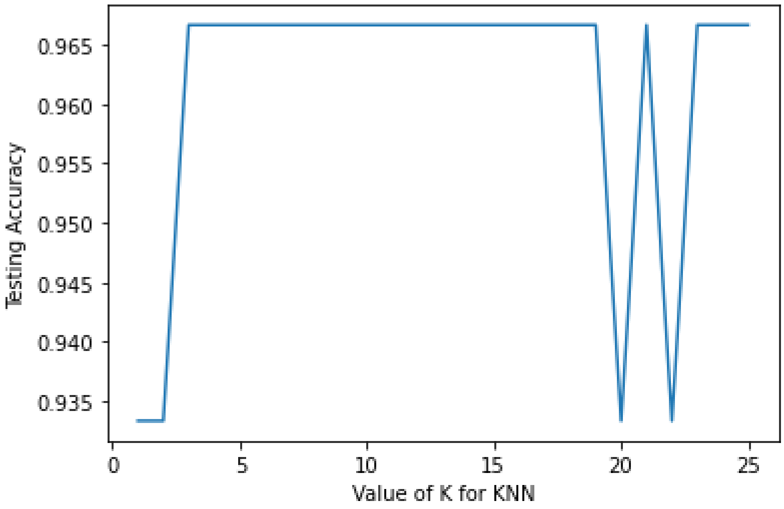
In [21]:

```
import matplotlib.pyplot as plt

#plot the relationship between K and the testing accuracy
plt.plot(KRange,[i for key,i in scores.items()])
plt.xlabel('Value of K for KNN')
plt.ylabel('Testing Accuracy')
```

Out[21]:

```
Text(0, 0.5, 'Testing Accuracy')
```



In [24]:

```
model = KNeighborsClassifier(n_neighbors=5)
model.fit(X,y)
```

```
C:\Users\Harsh\Anaconda3\lib\site-packages\ipykernel_launcher.py:2: DataConv
ersionWarning: A column-vector y was passed when a 1d array was expected. Pl
ease change the shape of y to (n_samples, ), for example using ravel().
```

Out[24]:

```
KNeighborsClassifier(algorithm='auto', leaf_size=30, metric='minkowski',
                     metric_params=None, n_jobs=None, n_neighbors=5, p=2,
                     weights='uniform')
```

In [26]:

```
xnew = [[1,2,3,4],[5,4,2,2]]

ypred = model.predict(xnew)

classes = {0:'setosa',1:'versicolor',2:'virginica'}
print(classes[ypred[0]])
print(classes[ypred[1]])
```

```
versicolor
setosa
```