In [1]:
```
import numpy as np
import matplotlib.pyplot as plt
import sklearn.datasets as dt
import sklearn.model_selection as ms
import sklearn.neural_network as nn
```

In [2]:
```
IRIS = dt.load_iris()
x = IRIS.data
y = IRIS.target
```

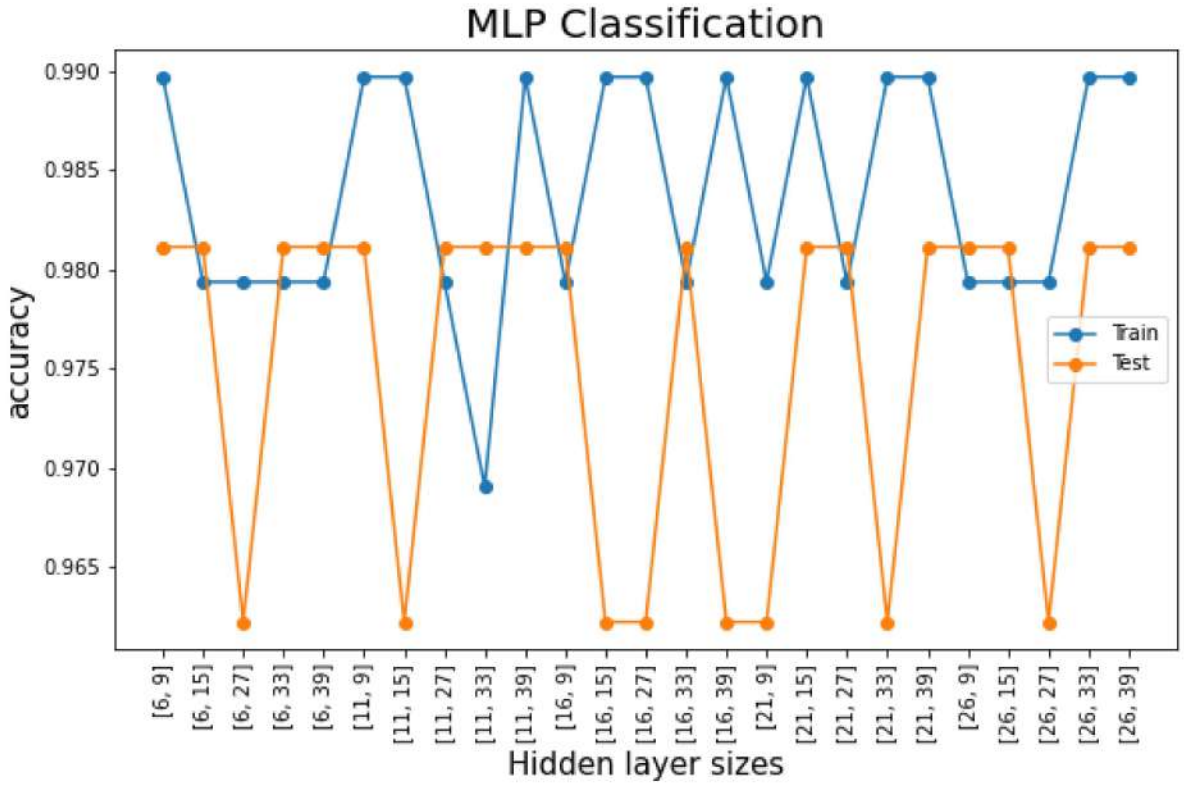
In [3]:
```
xtr, xte, ytr, yte = ms.train_test_split(x, y, train_size = 0.65, random_state = 35)
```

In [4]:
```
trAccuracy = []
teAccuracy = []
hidden_layer_sizes = []

for i in [6, 11, 16, 21, 26]:
    for j in [9, 15, 27, 33, 39]:
        NN = nn.MLPClassifier(hidden_layer_sizes = (i, j),
                              activation = "relu",
                              solver = "adam",
                              max_iter = 3000,
                              alpha = 0.01)
        NN.fit(xtr, ytr)
        trAccuracy.append(NN.score(xtr, ytr))
        teAccuracy.append(NN.score(xte, yte))
        hidden_layer_sizes.append([i, j])
```

In [9]:
```
plt.figure(figsize = (9.5, 5.5))
plt.plot(trAccuracy, label = "Train", marker = "o")
plt.plot(teAccuracy, label = "Test", marker = "o")
plt.xticks(range(0, len(hidden_layer_sizes)), hidden_layer_sizes, rotation = 90)
plt.title("MLP Classification", fontsize = 20)
plt.xlabel("Hidden layer sizes", fontsize = 15)
plt.ylabel("accuracy", fontsize = 15)
plt.legend()
```

Out[9]:
```
<matplotlib.legend.Legend at 0x17c805675b0>
```


MLP Classification
accuracy
0.990
0.985
0.980
0.975
0.970
0.965
Train
Test
[6, 9]
[6, 15]
[6, 27]
[6, 33]
[6, 39]
[11, 9]
[11, 15]
[11, 27]
[11, 33]
[11, 39]
[16, 9]
[16, 15]
[16, 27]
[16, 33]
[16, 39]
[21, 9]
[21, 15]
[21, 27]
[21, 33]
[21, 39]
[26, 9]
[26, 15]
[26, 27]
[26, 33]
[26, 39]
Hidden layer sizes

In [ ]: