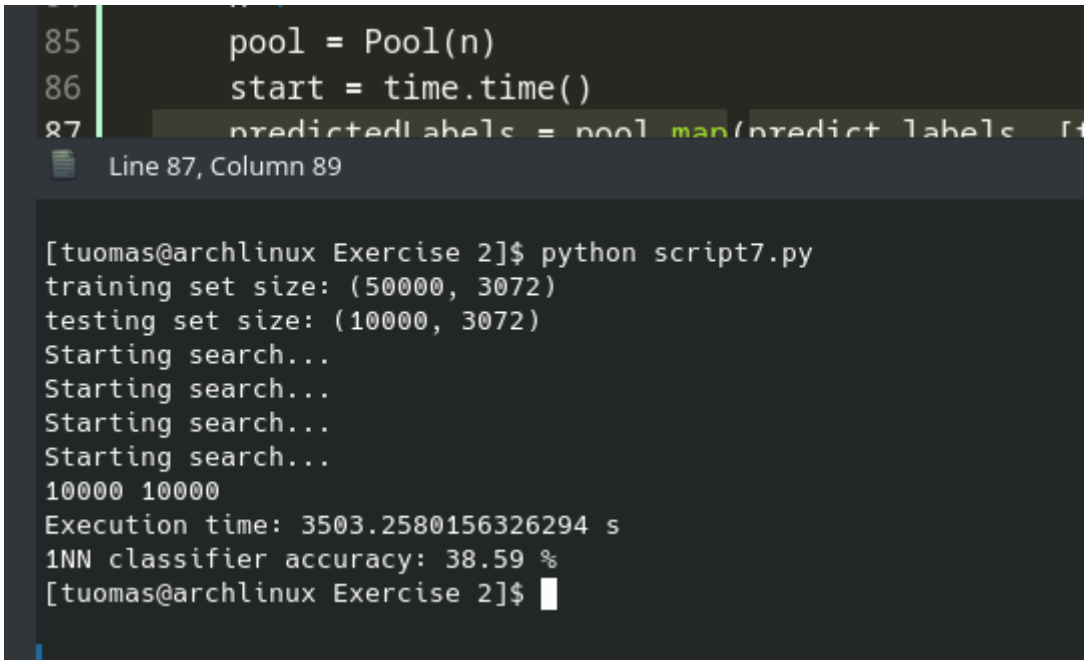
Task 4. **CIFAR-10 - 1-NN** classifier (20 points)

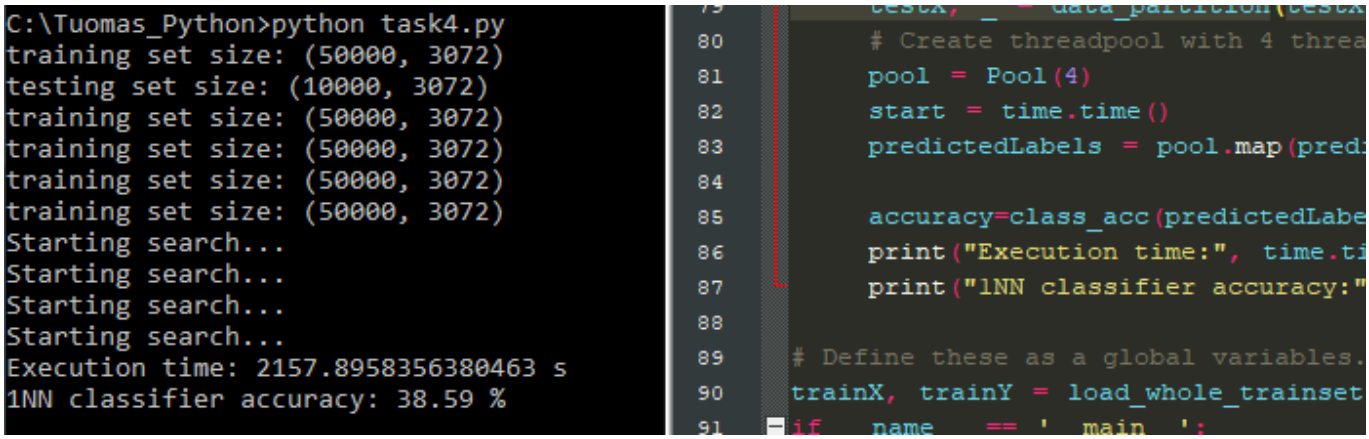
Here are some stats from my implementation:

Distance measure used: Absolute difference

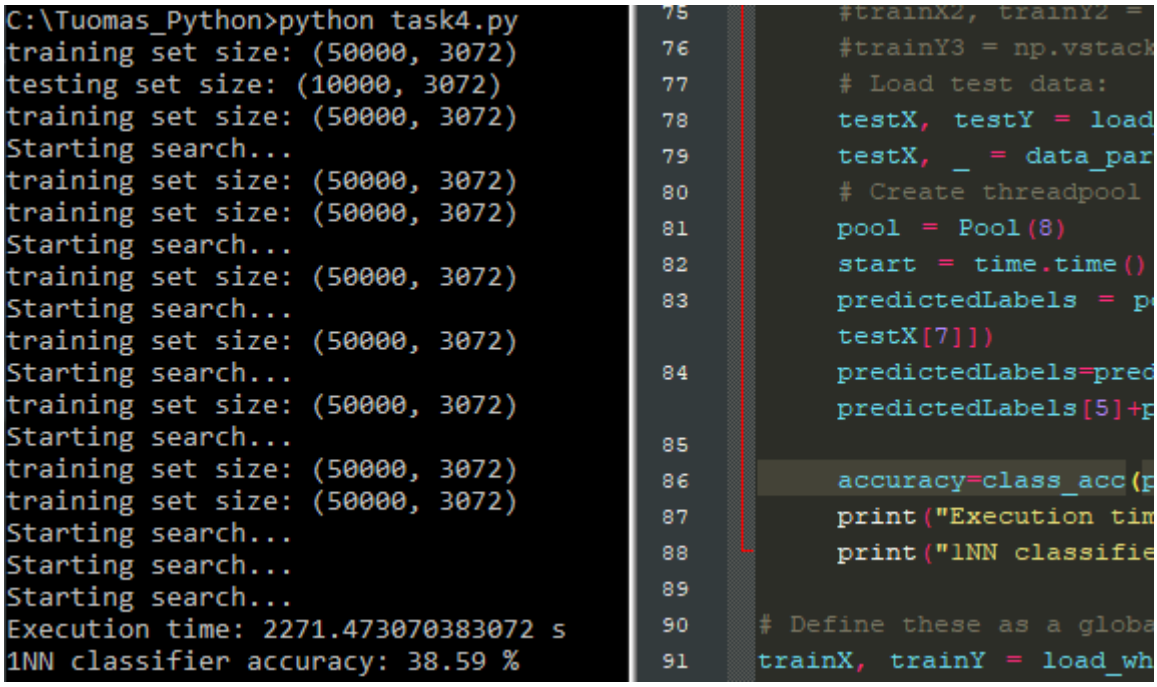
CPU used	**Accuracy**	**Execution time**	**Threads**	**Picture**
Intel i5-3230M (4) @ 3.200GHz	38.59%	3503s ≈ 1h	4	1
Intel Core i5-8400 @ 2.80GHz	38.59%	2158s ≈ 36 min	4	2
Intel Core i5-8400 @ 2.80GHz	38.59%	2271s ≈ 38 min	8	3



Picture 1.



Picture 2.


C:\Tuomas_Python>python task4.py
training set size: (50000, 3072)
testing set size: (10000, 3072)
training set size: (50000, 3072)
Starting search...
training set size: (50000, 3072)
training set size: (50000, 3072)
Starting search...
training set size: (50000, 3072)
Starting search...
training set size: (50000, 3072)
Starting search...
training set size: (50000, 3072)
Starting search...
training set size: (50000, 3072)
training set size: (50000, 3072)
Starting search...
Starting search...
Starting search...
Execution time: 2271.473070383072 s
1NN classifier accuracy: 38.59 %
#trainX2, trainY2 =
#trainY3 = np.vstack
Load test data:
testX, testY = load
testX, _ = data_par
# Create threadpool
pool = Pool(8)
start = time.time()
predictedLabels = p
testX[7]])
predictedLabels=pred
predictedLabels[5]+p
accuracy=class_acc(p
print("Execution tim
print("1NN classifie
Define these as a globa
trainX, trainY = load_wh

Picture 3.