CSE 7345

Quest 8: Regression and Matrix Transpose Analysis

Due: Sunday Nov 11 at midnight

Matrix transpose is a time-consuming function. Consider three approaches to computing the transpose:

A. Simple

```
m = [[1,2],[3,4],[5,6]]
for row in m :
    print(row)
rez = [[m[j][i] for j in range(len(m))] for i in range(len(m[0]))]
print("\n")
for row in rez:
    print(row)
```

B. zip

```
# Transpose with zip
# If a=[[1,2,3],[4,5,6],[7,8,9]]
# then  >>> zip(a[0],a[1],a[2])
              [(1, 4, 7), (2, 5, 8), (3, 6, 9)]
# To use zip when you don't know the size use: zip(*a)

matrix=[(1,2,3),(4,5,6),(7,8,9),(10,11,12)]
for row in matrix:
    print(row)
print("\n")
t_matrix = zip(*matrix)
for row in t_matrix:
    print (row )
```

C. numpy

```
import numpy
m2 = numpy.transpose(m1)
```

Perform timing experiments to determine which of the approaches is faster.
Test for the following matrix sizes.
Small: 200 x 200
Medium: 2000 x 2000
Large: 20000 x 20000
Really Large: ?? -- note that 50000x50000 will crash a PC due to out of memory error

DO: Prepare summary of your experiments
Include:

- Graph that clearly shows the results of your experiments
- Write up your results. What are the implications?
- What is the time complexity of matrix transposition?
- Do your results confirm the theory?

Submit a PDF to Canvas