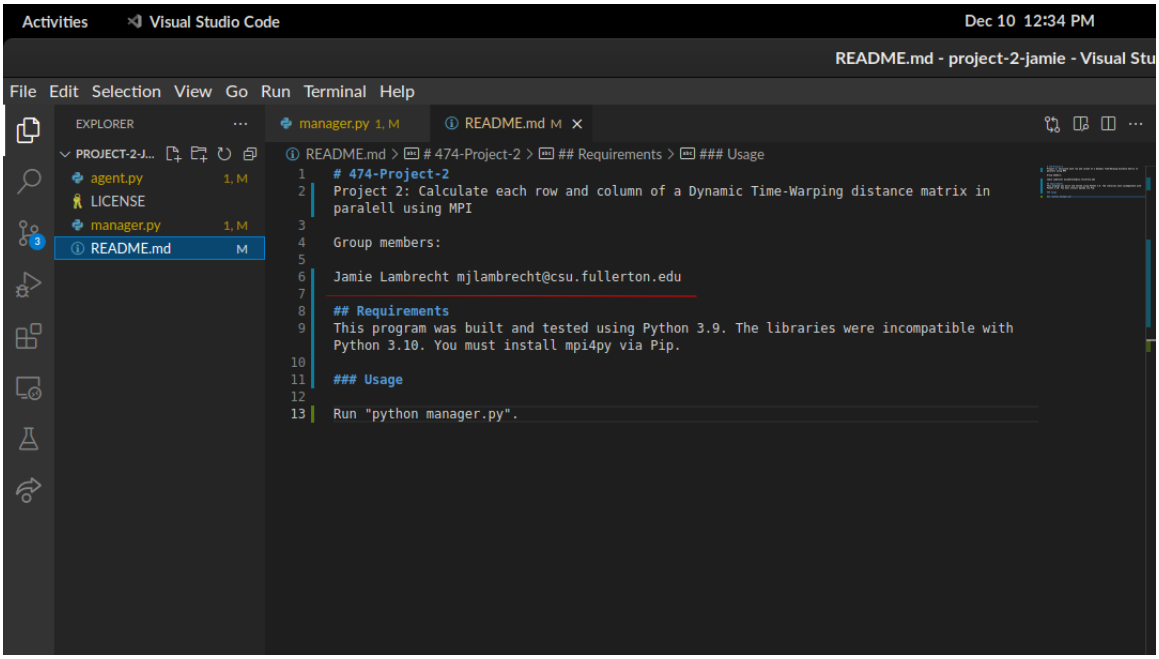
CPSC 474-01 Project 2

Jamie Lambrecht
mjlambrecht@csu.fullerton.edu

December 10, 2021

1 Screenshot of names in README.md



2 Overview

This is a distributed program with a manager process and two agent processes. The purpose of this program is to calculate the "dynamic time-warping distance". This is an algorithm for finding an optimal alignment between time-series with different frequencies (e.g. two vectors of point coordinates representing similar motion at different speed). This kind of data can be used for classification in machine learning algorithms, such as identifying hand gestures, etc.

Each agent has a starting point variable based on the subsequence (row or column) index number that is sent to the other on writing to the first cell in its subset. There is a critical section when an agent begins a new iteration. It must send a request for its starting point from a queue held by the manager process.

If the manager process receives that request it will send a message to initiate blocking in the other agents routine so that it does not continue until the first agent has received its starting point and begun its iteration.

To run the program, clone the source repository and run manager.py using the Python interpretor. You must install the correct version of python (3.9) as well mpi4py and any of its dependencies (mpich or openmpi, gcc, etc).

3 Psuedocode

3.1 Manager Process

```
begin manager process
spawn 2 agent processes

initialize data (two lists of tuples containing x and y coordinates of points)
create an M x N matrix cost_matrix.
for i in range(M):
  for j in range(N):
    cost_matrix[i,j] = (A[i][0] - B[j][0])**2 + (A[i][1] - B[j][1])**2

broadcast the matrix dimensions and cost_matrix to the agent processes
create M x N matrix dtw_matrix
copy top left cell of cost_matrix to dtw_matrix

while not complete:
  if update_A:
      send current row and column indices, current row and left neighbor to agent 0
  if update_B:
      send current row and column indices, current column and up neighbor to agent 1

  wait for agents to receive data
  increment row and column indices
  Receive processed data from agents
  set dtw_matrix row and column from received data

  if all rows and columns are complete:
    complete = True

print dtw_matrix
```

3.2 Agent Process

```
Receive broadcasted cost matrix and dimensions

receive current row and column indices
receive current row/column and left/up neighbor

for i in previous row or column:
  current[i] = min(previous[i], left_neighbor, previous[i - 1]) + cost_matrix[column index, row index]
  left or upneighbor = current[i]

send computed row or column back to manager process
```

4 Screenshot of program output

I was unfortunately unable to get this code working before the due date. Below is the output so far. As you can see, the first row and column are computed by the agent processes and passed back to the manager. There are a lot of indices and sizes being managed and many of them have issues of being swapped or "off-by-one". I will be trying to fix this code but I will simply not be able to fix it by midnight on Friday Dec. 10.

```
project-2-jamie on  main [!] via 🐍 v3.10.0 took 2s
❯ /bin/python3.9 /home/jamie/Repos/mpi-project/manager.py
Remote group size:2
This program requires exactly 3 processes, one manager and two agents. I am the manager.
About to send matrix dimensions (M x N): [4 3]
1
0
Length of returned row: 2
1 / 2 just received: [4 3]
2 / 2 just received: [4 3]
Length of returned column: 3
Just received:
[[ 2 20  8]
 [10  4 16]
 [26  4  8]
 [25 17  1]]
Just received:
[[ 2 20  8]
 [10  4 16]
 [26  4  8]
 [25 17  1]]
Processing for previous row: [0 0]
Left neighbor: 2
current_column_index: 0
current_row_index: 0
Finished processing current row: [22 30]
Processing for previous column: [0 0 0]
Up neighbor: 2
current_column_index: 0
current_row_index: 0
Finished processing current column: [12 38 63]
Manager received:
[22 30]
[12 38 63]
DTW Matrix:
[[ 2 22 30]
 [12  0  0]
 [38  0  0]
 [63  0  0]]
Current row [22 30]
Current column [12 38 63]
Length of returned row: 1
Length of returned column: 2
Processing for previous row: [22 30]
Left neighbor: 12
current_column_index: 1
current_row_index: 1
Processing for previous column: [12 38 63]
Up neighbor: 22
current_column_index: 1
current_row_index: 1
Finished processing current row: [16 20]
Finished processing current column: [16 16 20]
Traceback (most recent call last):
  File "/home/jamie/Repos/mpi-project/manager.py", line 94, in <module>
    comm.Recv(current_row, source = 0, tag=8)
  File "mpi4py/MPI/Comm.pyx", line 299, in mpi4py.MPI.Comm.Recv
mpi4py.MPI.Exception: MPI_ERR_TRUNCATE: message truncated
```