

CPSC 535 Advanced Algorithms

Project 2: It's a Small World

Instructor: Prof. Doina Bein

Submission date: 9th December 2022

Priya Keshri
[885191452]
priyakeshri@csu.fullerton.edu

Shrinivas Pravin Patil
[885212043]
pshrinivas264@csu.fullerton.edu

Summary:

Python3 version 3.10 is used to implement the project. Install the Python compiler and editor as 'PyCharm'. The code takes input from a text file named input.txt. First line of the text file represents the total number of test cases. From the next line each line contains a number of casts in each test case followed by casts on each line. The expected output is the shortest connection between cast[0] and cast[1] and common actor or cast.

Pseudocode:

- 1) Read all the test cases from the text file.
 - 2) Create list of casts named cast=[[]]*n
 - 3) Store each cast of actors by preprocessing each line by splitting and removing new line and spaces.
 - 4) if cast[0] & cast[1]:
 conn=1
 actor=list(set(cast[0] & cast[1]))
 return "shortest connection={},actor/actors={}".format(conn,cast[i])
- For i goes from 2 to n:
 If set(cast[i]) & set(cast[0]):
 If set(cast[i]) & set(cast[1]):
 conn=2
 Return "Shortest connection = {}, conn={}".format(conn,cast[i])
- Return "shortest connection > 2 or no connection"

Code:

```
import os
import sys

def find_connect(n,cast):
    conn = 0
    try:
        if set(cast[0])&set(cast[1]):      # Convert the list into set and find common actor using
intersection
            conn = 1
            actor = list(set(cast[0])&set(cast[1]))
            return "shortest connection = {} , actor/actors = {}".format(conn,actor)
        for i in range(2,n):              # Check for cast 2 to cast n for shortest connection
            if set(cast[i])&set(cast[0]):
                if set(cast[i])&set(cast[1]):
                    conn = 2
                    return "shortest connection = {} , cast = {}".format(conn,cast[i])
        return "shortest connection > 2 or no connection"
    except:
        return "Please try it again!!"

if __name__ == '__main__':
    with open('input.txt','r') as fl: # Reading Text file
        cases = int(fl.readline())    # Reading the first line which number of test cases
```

```

while cases:
    n = int(fl.readline())    # Reading the length of list
    cast = [[]]*n           # Creating list of casts
    for i in range(n):
        cast[i] = list(map(str.strip,fl.readline().replace("\n','").split(','))) # Processing each line and
store in the list.
    #print(cast)
    print(find_connect(n,cast)) # Calling the function
    cases-=1

```

The screenshot shows a GitHub repository page for 'project-1---it-s-a-small-world-shrinivas-priya'. The repository is private and has 0 stars, 0 forks, and 0 watchers. The main branch is 'main' with 2 branches and 0 tags. The repository was created by GitHub Classroom. The README file is visible, showing the title '535-project2' and the content 'It's a small world'. The repository also has a 'LICENSE' file and a 'README.md' file. The repository was created by GitHub Classroom.

Repository Details:

- Repository: CSUF-CPSC-Bein-Fall-2022 / project-1---it-s-a-small-world-shrinivas-priya (Private)
- Watch: 0
- Fork: 0
- Star: 0
- Code: 0
- Issues: 0
- Pull requests: 0
- Actions: 0
- Projects: 0
- Security: 0
- Insights: 0

Files:

- Project 2: Add files via upload (5 minutes ago)
- LICENSE: Initial commit (2 hours ago)
- README.md: Update README.md (2 hours ago)

README.md:

535-project2

It's a small world

Group members:

- Priya Keshri priyakeshri@csu.fullerton.edu
- Shrinivas Pravin Patil pshrinivas264@csu.fullerton.edu

About:

project-1---it-s-a-small-world-shrinivas-priya created by GitHub Classroom

Releases:

No releases published
[Create a new release](#)

Packages:

No packages published
[Publish your first package](#)

Languages:

Test Case:

Input:

```
main.py x input.txt
Users > JanhviGuha > Desktop > priya_project2 > input.txt
1 3
2 6
3 Carrie-Anne Moss, Gloria Foster, Hugo Weaving, Joe Pantoliano, Keanu Reeves, Laurence Fishburne, Marcus Chong
4 Andre Braugher, Beau Garrett, Chris Evans, Doug Jones, Ioan Gruffudd, Jessica Alba, Julian McMahon, Kerry Washington, Laurence Fishburne, Michael Chiklis
5 Ewan McGregor, Ian McDiarmid, Jake Lloyd, Liam Neeson, Natalie Portman
6 Geoffrey Rush, Jack Davenport, Johnny Depp, Jonathan Pryce, Keira Knightley, Orlando Bloom
7 Angela Bassett, Chadwick Boseman, Danai Gurira, Daniel Kaluuya, Forest Whitaker, Letitia Wright, Lupita Nyong'o, Martin Freeman, Michael B. Jordan, Sterling K. Brown
8 Andrew Borba, Anne Hathaway, Bill Irwin, Casey Affleck, Collette Wolfe, David Oyelowo, Francis X. McCarthy, Jessica Chastain, John Lithgow, Matthew McConaughey, Mich
9 7
10 Carrie-Anne Moss, Gloria Foster, Hugo Weaving, Joe Pantoliano, Keanu Reeves, Laurence Fishburne, Marcus Chong
11 Andrew Borba, Anne Hathaway, Bill Irwin, Casey Affleck, Collette Wolfe, David Oyelowo, Francis X. McCarthy, Jessica Chastain, John Lithgow, Matthew McConaughey, Mich
12 Geoffrey Rush, Jack Davenport, Johnny Depp, Jonathan Pryce, Keira Knightley, Orlando Bloom
13 Angela Bassett, Chadwick Boseman, Danai Gurira, Daniel Kaluuya, Forest Whitaker, Letitia Wright, Lupita Nyong'o, Martin Freeman, Michael B. Jordan, Sterling K. Brown,
14 Abraham Attah, Asa Butterfield, Anne Hathaway, Chloe Grace Moretz, Daniel Radcliffe, Jeff Goldblum, Keanu Reeves, Tom Holland
15 Andre Braugher, Beau Garrett, Chris Evans, Doug Jones, Ioan Gruffudd, Jessica Alba, Julian McMahon, Kerry Washington, Laurence Fishburne, Michael Chiklis
16 Ewan McGregor, Ian McDiarmid, Jake Lloyd, Liam Neeson, Natalie Portman
17 7
18 Ewan McGregor, Ian McDiarmid, Jake Lloyd, Liam Neeson, Natalie Portman
19 Andrew Borba, Anne Hathaway, Bill Irwin, Casey Affleck, Collette Wolfe, David Oyelowo, Francis X. McCarthy, Jessica Chastain, John Lithgow, Matthew McConaughey, Mich
20 Geoffrey Rush, Jack Davenport, Johnny Depp, Jonathan Pryce, Keira Knightley, Orlando Bloom
21 Angela Bassett, Chadwick Boseman, Danai Gurira, Daniel Kaluuya, Forest Whitaker, Letitia Wright, Lupita Nyong'o, Martin Freeman, Michael B. Jordan, Sterling K. Brown,
22 Abraham Attah, Asa Butterfield, Anne Hathaway, Chloe Grace Moretz, Daniel Radcliffe, Jeff Goldblum, Keanu Reeves, Tom Holland
23 Andre Braugher, Beau Garrett, Chris Evans, Doug Jones, Ioan Gruffudd, Jessica Alba, Julian McMahon, Kerry Washington, Laurence Fishburne, Michael Chiklis
24 Carrie-Anne Moss, Gloria Foster, Hugo Weaving, Joe Pantoliano, Keanu Reeves, Laurence Fishburne, Marcus Chong

PROBLEMS OUTPUT JUPYTER DEBUG CONSOLE
shortest connection = 1 , actor/actors = ['Laurence Fishburne']
shortest connection = 2 , cast = ['Abraham Attah', 'Asa Butterfield', 'Anne Hathaway', 'Chloe Grace Moretz', 'Daniel Radcliffe', 'Jeff Goldblum', 'Keanu Reeves', 'Tom Holland']
shortest connection > 2 or no connection

[Done] exited with code=0 in 0.062 seconds
Ln 1, Col 1 Spaces: 4 UTF-8 LF Plain Text
```

Output:

```
main.py x input.txt
Users > JanhviGuha > Desktop > priya_project2 > main.py > ...
1 import os
2 import sys
3
4 def find_connect(n,cast):
5     conn = 0
6     try:
7         if set(cast[0])&set(cast[1]): # Convert the list into set and find common actor using intersection
8             conn = 1
9             actor = list(set(cast[0])&set(cast[1]))
10            return "shortest connection = {} , actor/actors = {}".format(conn,actor)
11        for i in range(2,n): # Check for cast 2 to cast n for shortest connection
12            if set(cast[i])&set(cast[0]):
13                if set(cast[i])&set(cast[1]):
14                    conn = 2
15                    return "shortest connection = {} , cast = {}".format(conn,cast[i])
16            return "shortest connection > 2 or no connection"
17        except:
18            return "Please try it again!!"
19
20 if __name__ == '__main__':
21     with open('input.txt','r') as fl: # Reading Text file
22         cases = int(fl.readline()) # Reading the first line which number of test cases
23         while cases:
24             n = int(fl.readline()) # Reading the length of list
25             cast = [list]*n # Creating list of casts
26             for i in range(n):
27                 cast[i] = list(map(str.strip,fl.readline().replace('\n','').split(','))) # Processing each line and store in the list.
28             #print(cast)
29             print(find_connect(n,cast)) # Calling the function
30             cases-=1
31
```

```
PROBLEMS OUTPUT JUPYTER DEBUG CONSOLE
shortest connection = 1 , actor/actors = ['Laurence Fishburne']
shortest connection = 2 , cast = ['Abraham Attah', 'Asa Butterfield', 'Anne Hathaway', 'Chloe Grace Moretz', 'Daniel Radcliffe', 'Jeff Goldblum', 'Keanu Reeves', 'Tom Holland']
shortest connection > 2 or no connection

[Done] exited with code=0 in 0.062 seconds
Ln 1, Col 10 Spaces: 4 UTF-8 LF Python 3.9.7 64-bit
```

```
PROBLEMS  OUTPUT  JUPYTER  DEBUG CONSOLE
[Running] python -u "/Users/JanhviGuha/Desktop/priya_project2/main.py"
shortest connection = 1 , actor/actors = ['Laurence Fishburne']
shortest connection = 2 , cast = ['Abraham Attah', 'Asa Butterfield', 'Anne Hathaway', 'Chloe Grace Moretz', 'Daniel Radcliffe', 'Jeff Goldblum', 'Keanu Reeves', 'Tom Holland']
shortest connection > 2 or no connection

[Done] exited with code=0 in 0.062 seconds
```

main* 0 0 0 Ln 31, Col 5 Spaces: 4 UTF-8 LF Python

Time Complexity: $O(n)$

Reason: In this algorithm, whenever we call the `find_connect()`, the while loop will run for $n-2$ times in the worst case where n is the number of cities.

Space Complexity: $O(n*m)$

Reason: In this algorithm worst case space complexity will be $n*m$ where n = total number of casts and m = Total number of actors in each cast.