中山大學
SUN YAT-SEN UNIVERSITY

高级编程技术实验报告

实验二：最短路径

数据科学与计算机学院　17大数据与人工智能

17341015　陈鸿峥

一、 问题描述及求解思路

1. 创建数据结构

对于WeightedEdge的初始化如下

- __init__：调用父类构造函数Edge.__init__进行初始化
- get_total_distance和get_outdoor_distance：均直接返回值
- __str__：类似Edge类的字符转化方法

对于Digraph的初始化如下

- add_node：先用has_node判断结点是否在结点集合中，如果不存在则报错raise ValueError；否则添加结点到集合中
- add_edge：同样先判断结点是否在结点集合中，如果存在则判断是否新加结点，如果是新加结点则创建一个新列表，否则直接插入原有列表中

2. 创建校园地图

下面问题的答案在ps2.py中也可以找到

a. 每一个图结点代表校园里的一栋大楼，每一条边代表一栋大楼和另一栋大楼相邻（有路可走），距离/边权则代表一栋大楼到另一栋大楼的距离。

b. load_map：先创建一个空对象g=Digraph()，利用with open(map_filename,"r") as file打开文件，防止文件不存在仍继续执行。然后遍历文件的每一行，通过split函数按照空格分割，同时可以如下用四元组直接赋值。

```
(src,dst,tot_dist,outdoor_dist) = line.split(’ ’)
```

判断图g中是否存在源结点和汇结点，如果不存在则添加。然后创建有权边WeightedEdge，通过add_edge添加入图g中。

c. 如下测试输出是否合规。

```
school_map = load_map("test_load_map.txt")
print(school_map)
```

3. 用优化DFS寻找最短路径

(i) 目标函数

- 目标函数：最小化从源结点到汇结点的距离
- 限制条件：出校外的距离不能超过max_dist_outdoors

(ii) 实施最优路径

伪代码已经在题目描述中给出，下面是完整代码，原题注释已删除。

```
def get_best_path(digraph, start, end, path, max_dist_outdoors, best_dist,
                  best_path):
    # not valid nodes
    if not (digraph.has_node(Node(start)) and digraph.has_node(Node(end))):
        raise ValueError
    # recursion termination
    elif start == end:
        if path[1] < best_dist:
            best_path = copy.deepcopy(path[0])
            best_dist = path[1]
    # get deeper
    else:
        for edge in digraph.get_edges_for_node(Node(start)):
            dst = edge.get_destination()
            if dst in path[0]: # avoid cycles
                continue
            tmp_path = copy.deepcopy(path)
            tmp_path[0].append(dst)
            tmp_path[1] += edge.get_total_distance()
            tmp_path[2] += edge.get_outdoor_distance()
            if tmp_path[2] > max_dist_outdoors: # not satisfy the constraint
                continue
            if tmp_path[1] > best_dist: # cut!!! greatly reduce time
                continue
            (tmp_best_path,tmp_best_dist) = get_best_path(digraph,dst,end,
                tmp_path,max_dist_outdoors,best_dist,best_path)
            if tmp_best_dist < best_dist: # update best path
                best_path = tmp_best_path
                best_dist = tmp_best_dist
    return (best_path,best_dist)
```

这里有几个点需要注意：

- 每次对path进行修改之前都记得要进行**深拷贝**，仅仅是数组的.copy是不够的，因为path是一个嵌套列表。故这里我采用了copy.deepcopy的方式进行深拷贝

- 为了避免成环的情况的情况，先判断新结点是否已经在路径上，如果是则直接跳过
- 对于新加入的边，如果总长度已经超过最优长度，则后面的搜索都可以被裁剪掉，这将大大缩短运行时间（见图3和图4的比较）
- 每次迭代完返回都要将值赋给临时变量tmp_best_path和tmp_best_dist，在当前层所有结点都遍历完后才返回最优路径

(iii) 实施有向DFS

这里直接调用上一问的get_best_path函数即可。代码如下，原题注释已删除。

```
def directed_dfs(digraph, start, end, max_total_dist, max_dist_outdoors):
    (best_path,best_dist) = get_best_path(digraph,start,end,[[start],0,0],
       ↪ max_dist_outdoors,max_total_dist,[])
    if best_path == []:
        raise ValueError
    else:
        return best_path
```

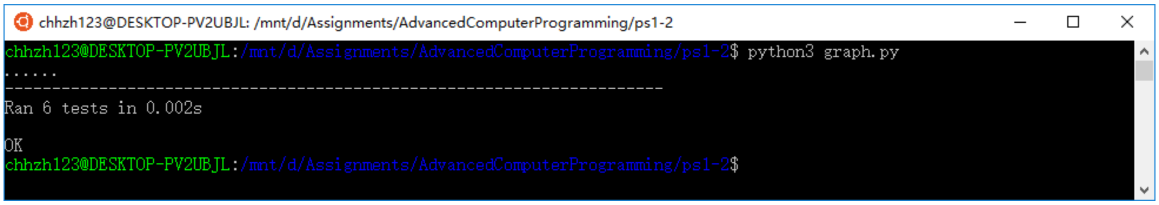
注意start要先存入初始列表中，然后将最优路径长度设为最大路径长。如果没有比最大路径长更小的路径，则对应的best_path也会为空，进而报错raise ValueError。

二、 代码

代码实施及注释请见附件graph.py、ps2.py。自己写的测试样例见test_load_map.txt。

三、 运行截图

实验运行结果如下面几幅图片所示。

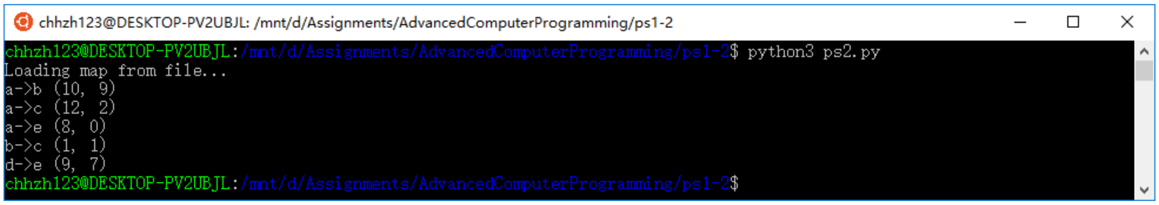


chhzh123@DESKTOP-PV2UBJL: /mnt/d/Assignments/AdvancedComputerProgramming/ps1-2

```
chhzh123@DESKTOP-PV2UBJL:/mnt/d/Assignments/AdvancedComputerProgramming/ps1-2$ python3 graph.py
......
----------------------------------------------------------------------
Ran 6 tests in 0.002s

OK
chhzh123@DESKTOP-PV2UBJL:/mnt/d/Assignments/AdvancedComputerProgramming/ps1-2$
```

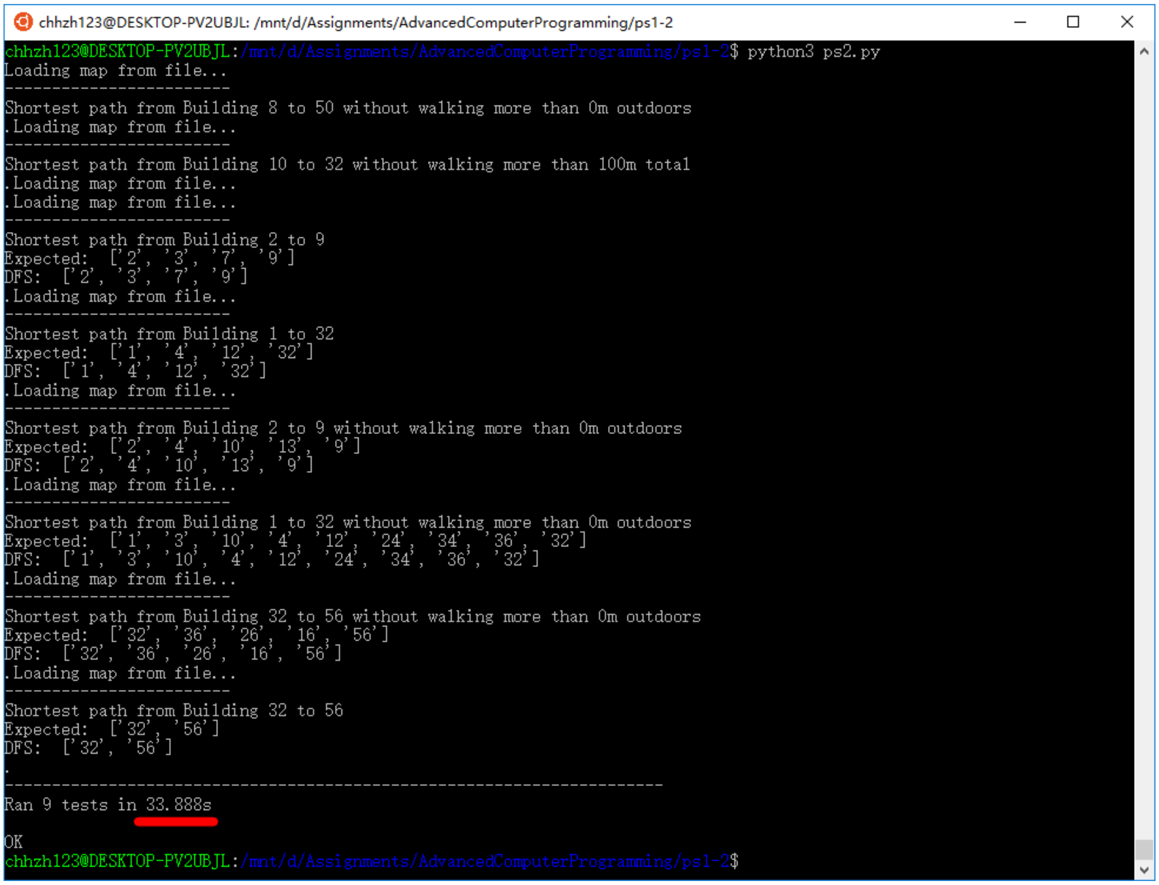
图 1: 问题1结果，测试样例全通过



chhzh123@DESKTOP-PV2UBJL: /mnt/d/Assignments/AdvancedComputerProgramming/ps1-2

```
chhzh123@DESKTOP-PV2UBJL:/mnt/d/Assignments/AdvancedComputerProgramming/ps1-2$ python3 ps2.py
Loading map from file...
a->b (10, 9)
a->c (12, 2)
a->e (8, 0)
b->c (1, 1)
d->e (9, 7)
chhzh123@DESKTOP-PV2UBJL:/mnt/d/Assignments/AdvancedComputerProgramming/ps1-2$
```

图 2: 问题2结果，测试样例输出正常

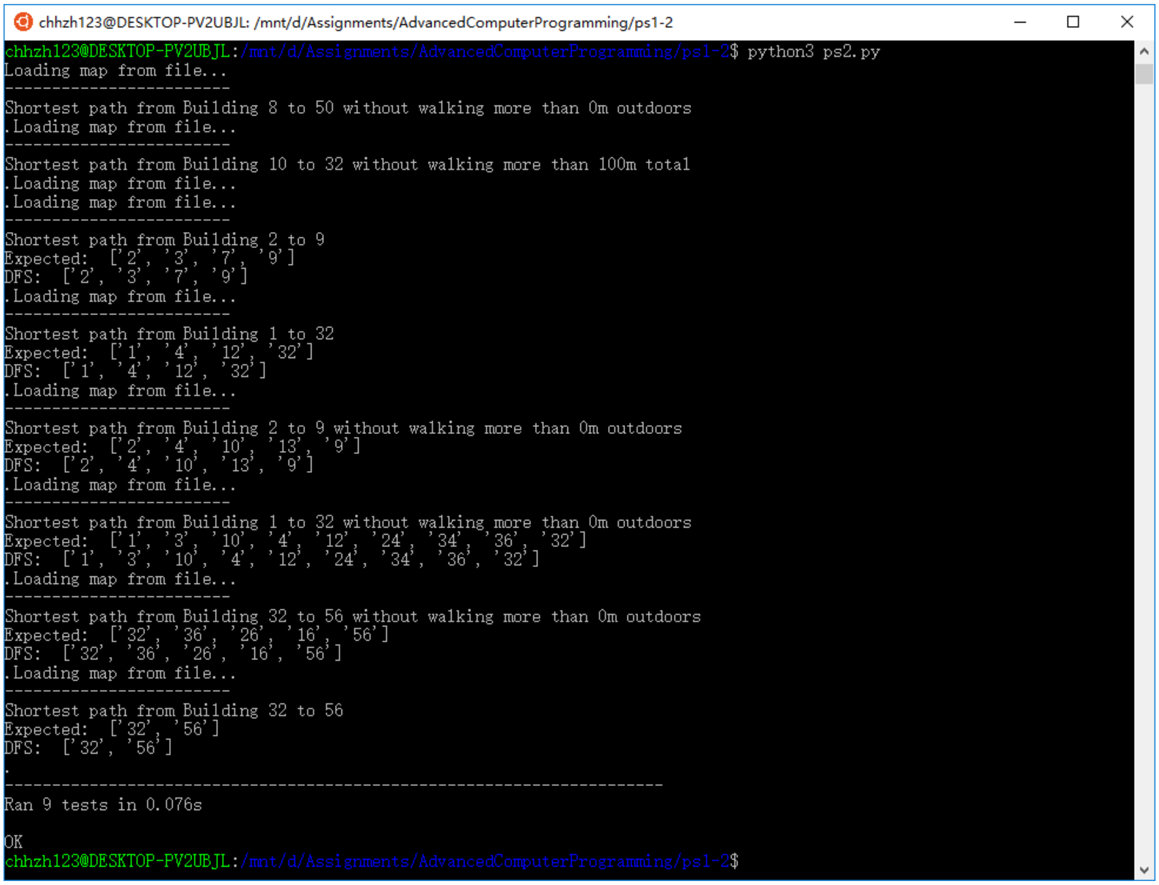


chhzh123@DESKTOP-PV2UBJL: /mnt/d/Assignments/AdvancedComputerProgramming/ps1-2

```
chhzh123@DESKTOP-PV2UBJL:/mnt/d/Assignments/AdvancedComputerProgramming/ps1-2$ python3 ps2.py
Loading map from file...
------------------------
Shortest path from Building 8 to 50 without walking more than 0m outdoors
.Loading map from file...
------------------------
Shortest path from Building 10 to 32 without walking more than 100m total
.Loading map from file...
.Loading map from file...
------------------------
Shortest path from Building 2 to 9
Expected:  ['2', '3', '7', '9']
DFS:  ['2', '3', '7', '9']
.Loading map from file...
------------------------
Shortest path from Building 1 to 32
Expected:  ['1', '4', '12', '32']
DFS:  ['1', '4', '12', '32']
.Loading map from file...
------------------------
Shortest path from Building 2 to 9 without walking more than 0m outdoors
Expected:  ['2', '4', '10', '13', '9']
DFS:  ['2', '4', '10', '13', '9']
.Loading map from file...
------------------------
Shortest path from Building 1 to 32 without walking more than 0m outdoors
Expected:  ['1', '3', '10', '4', '12', '24', '34', '36', '32']
DFS:  ['1', '3', '10', '4', '12', '24', '34', '36', '32']
.Loading map from file...
------------------------
Shortest path from Building 32 to 56 without walking more than 0m outdoors
Expected:  ['32', '36', '26', '16', '56']
DFS:  ['32', '36', '26', '16', '56']
.Loading map from file...
------------------------
Shortest path from Building 32 to 56
Expected:  ['32', '56']
DFS:  ['32', '56']
.
----------------------------------------------------------------------
Ran 9 tests in 33.888s

OK
chhzh123@DESKTOP-PV2UBJL:/mnt/d/Assignments/AdvancedComputerProgramming/ps1-2$
```

图 3: 问题3结果，测试样例全通过。没有剪枝，相当于枚举所有情况，速度非常慢


chhzh123@DESKTOP-PV2UBJL: /mnt/d/Assignments/AdvancedComputerProgramming/ps1-2

```
chhzh123@DESKTOP-PV2UBJL:/mnt/d/Assignments/AdvancedComputerProgramming/ps1-2$ python3 ps2.py
Loading map from file...
------------------------
Shortest path from Building 8 to 50 without walking more than 0m outdoors
.Loading map from file...
------------------------
Shortest path from Building 10 to 32 without walking more than 100m total
.Loading map from file...
.Loading map from file...
------------------------
Shortest path from Building 2 to 9
Expected:  ['2', '3', '7', '9']
DFS:  ['2', '3', '7', '9']
.Loading map from file...
------------------------
Shortest path from Building 1 to 32
Expected:  ['1', '4', '12', '32']
DFS:  ['1', '4', '12', '32']
.Loading map from file...
------------------------
Shortest path from Building 2 to 9 without walking more than 0m outdoors
Expected:  ['2', '4', '10', '13', '9']
DFS:  ['2', '4', '10', '13', '9']
.Loading map from file...
------------------------
Shortest path from Building 1 to 32 without walking more than 0m outdoors
Expected:  ['1', '3', '10', '4', '12', '24', '34', '36', '32']
DFS:  ['1', '3', '10', '4', '12', '24', '34', '36', '32']
.Loading map from file...
------------------------
Shortest path from Building 32 to 56 without walking more than 0m outdoors
Expected:  ['32', '36', '26', '16', '56']
DFS:  ['32', '36', '26', '16', '56']
.Loading map from file...
------------------------
Shortest path from Building 32 to 56
Expected:  ['32', '56']
DFS:  ['32', '56']
.
----------------------------------------------------------------------
Ran 9 tests in 0.076s

OK
chhzh123@DESKTOP-PV2UBJL:/mnt/d/Assignments/AdvancedComputerProgramming/ps1-2$
```

图 4: 问题3结果，测试样例全通过。运用剪枝，速度大幅提升