

سوال اول

هدف شما رساندن فروشنده به بازار با استفاده از الگوریتم‌های جستجو گرافی است. فروشنده در موقعیت اولیه T قرار دارد و می‌خواهد به بازار در موقعیت S برود. در این مسئله شما باید پس از دریافت ماتریسی مشابه نمونه زیر، با توجه به روش مسیریابی خواسته‌شده از بین روش‌های DFS، BFS و A* خروجی مناسب را چاپ کنید.

ورودی

در خط اول روش مسیریابی گفته می‌شود. روش‌های مسیریابی شامل dfs ، bfs و astar می‌شوند. سپس پس از آن نقشه شهر داده می‌شود که موانع با کاراکتر درصد % و مسیرها با کاراکتر فاصله مشخص شده‌اند.

توجه کنید خط شروع مسیر و خط آخر مسیر همواره یکی و تماماً % هستند. اندازه نقشه مشخص نیست و باید با استفاده از گزاره قبل متوجه پایان ورودی‌ها شوید. نقشه لزوماً مربعی نیست!

خروجی

خروجی برنامه‌ی شما باید حرکت‌های انجام شده از ابتدای بازی تا انتهای آن باشد. حرکت‌های مجاز شامل North ، East ، South و West هستند. در صورتی که مسیری از T به S وجود نداشت، باید پیغام مناسب مانند خروجی نمونه 2 چاپ کنید.

مثال

ورودی نمونه 1

```
dfs
%%%%
% T %
%%% %
```

%S %
%%%%%%%%

خروجی نمونه 1

East South South West West

توجه کنید که تنها در یک خط حرکات‌ها را بنویسید.

ورودی نمونه 2

astar
%%%%%%%%
%S% %
% % %
% %T%
%%%%%%%%

خروجی نمونه 2

This town be dard nemikhore!

ورودی نمونه 3

bfs
%%%%%%%%
% S%
% % % %
% %% %% %
% %% % %
% % % %
% %%%
%%%%%%%% % %
%%%%%%%% % % %

| | | |
|-----------|----|-------|
| % | % | % |
| % | %% | % % % |
| % | % | % |
| %%%%%%%% | | % |
| %T | %% | % |
| %%%%%%%%% | | |

خروجی نمونه 3

هر دو جواب زیر درست هستند که بستگی به ترتیب گسترش همسایگان در کد دارند. هر کدامشان را خروجی بدهید، تست را پاس می‌کنید.

East East East North North North North East East North North North North North

9

East East East North East East North North North North North North North North North

راهنمایی

برای گرفتن نقشه ورودی می‌توانید شبیه قطعه کد زیر عمل کنید.

```

1  def get_map():
2      map = []
3      first_row = input()
4      row = 0
5      line = first_row
6      is_last_line = False
7      start_node = None
8      target_node = None
9      while(True):
10         map.append([])
11         for column in range(len(first_row)):
12             # (row, column) corresponds to (y, x)

```

```

13         map[row].append(node(row, column))
14         if line[column] == '%':
15             map[row][column].is_wall = True
16         elif line[column] == ' ':
17             map[row][column].is_path = True
18         elif line[column] == 'T':
19             map[row][column].is_start = True
20             map[row][column].is_path = True
21             start_node = map[row][column]
22         elif line[column] == 'S':
23             map[row][column].is_target = True
24             map[row][column].is_path = True
25             target_node = map[row][column]
26     if not is_last_line:
27         line = input()
28         row += 1
29         if line == first_row:
30             is_last_line = True
31         continue
32     break
33     return map, start_node, target_node
34
35 search_method = input()
36 map, _, _ = get_map()

```

برای بررسی روش مسیریابی ترجیحاً به شکل زیر عمل کنید.

```

1     if 'bfs' in search_method:
2         # TODO bfs
3     elif 'dfs' in search_method:
4         # TODO dfs
5     elif 'astar' in search_method:
6         # TODO astar

```

سوال دوم

ابتدا فایل زیپ صورت سوال را از این [لینک](#) دانلود کنید. سپس نوت‌بوک `Coding-HW1-Genetic.ipynb` را باز کنید و موارد خواسته‌شده هر بخش را کامل نمایید. سپس نوت‌بوک کامل‌شده خود را آپلود نمایید. توجه کنید که نباید فایل `utils.py` را تغییر دهید.