

Artificial Intelligence

Project 1
Phase 1

Mohamed Ismail 201800595
Eslam Elsharkawy 201800473
Mohamed Hazem 201802190

WorkLoad

Mohamed Ismail:

1. Problem Formulation and Modeling

Eslam El Sharkawy:

1. Informed Search Algorithms
 - a. A*
 - b. GFS
 - c. Hill Climbing
 - d. Local Beam Search
 - e. Simulated Annealing

Mohamed Hazem:

1. Uninformed Search Algorithms
 - a. BFS
 - b. DFS
 - c. IDS
 - d. UCS

Requirements

- Problem formulation
 - Initial State
 - Goal State
 - State Space (main buildings, labs, research centres, operation departments, etc)
 - Path Cost
- Modeling Assumptions (bumps to slow down cars, pedestrian crossings, waiting areas, sun shields)
- Set of Actions
- Inputs & Outputs
- Methods (Successor Function)
 - Uninformed Search
 - Informed Search (heuristics)



Zewail City Map

State Space Representation



Zewail city has 6 gates and 8 buildings

One Stop Shop

Nano building

Helmy building

service building

Workshops

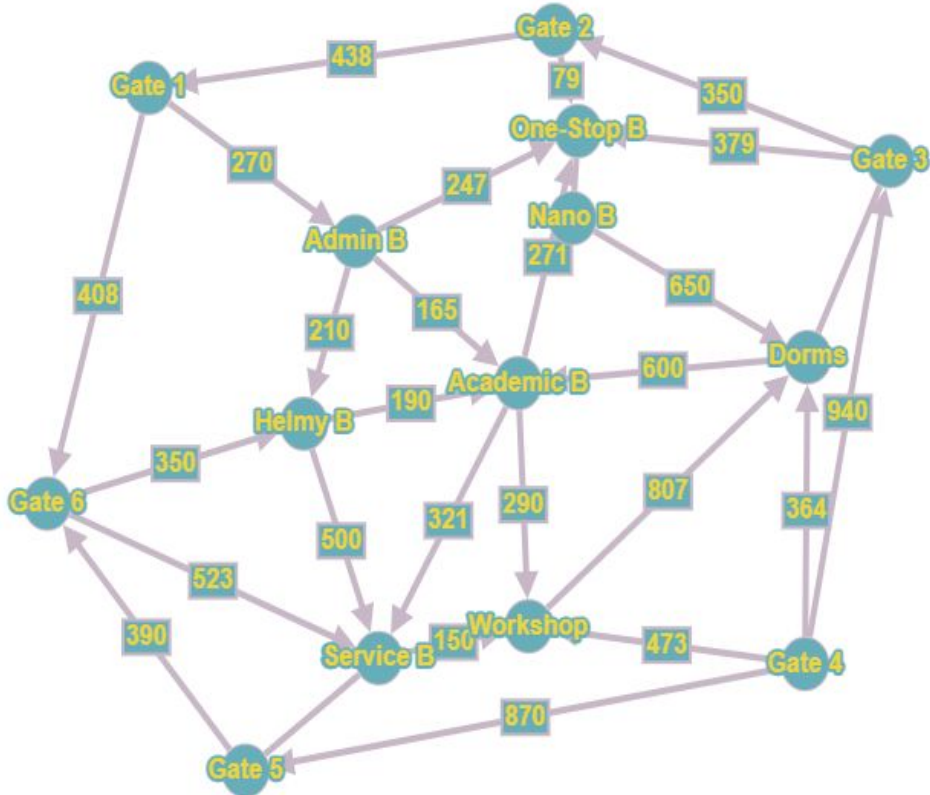
Dorms

Administrative building

The academic building (with different gates)

The aim is to start at any place in the city and reach another place, so the initial and goal states will be inputs from the user

Actions and Successors



On each place we have some available places to reach and that represents the actions and successors.

We can reach any place by different paths with different path costs. E.g,

The main gate → the administrative building= 270 m

The main gate → One Stop Shop = 470 m

The to administrative building → Helmy = 210 m

Problem Formulation

```
_ZC_ROAD_MAP = '''
#####
#####A#####
###===$$=====$$=====###
###=#####==#####==#####
###=#####==#####==#####
###=#####$$#####==#####
#F==#####==#####-----H---B##
##=#####==#####I-----##
##=#####==#####-----##
##=###-===xx=====---#####
'''
```

```
step_cost_map = {
    '=': 1,    # cars road
    '-': 2,    # passengers road
    'x': 3,    # crossing
    '$': 4,    # car slow down
}
```


Project Structure

```
.
├── dump
│   └── zc_map.py
├── lib
│   ├── __pycache__
│   │   ├── settings.cpython-310.pyc
│   │   └── utils.cpython-310.pyc
│   ├── settings.py
│   └── utils.py
├── main.py
├── models
│   ├── problem.py
│   ├── __pycache__
│   │   ├── problem.cpython-310.pyc
│   │   ├── tree.cpython-310.pyc
│   │   └── zc_map.cpython-310.pyc
│   ├── tree.py
│   ├── zc_map.py
│   └── settings.json
├── solvers
│   ├── a_star.py
│   ├── bfs.py
│   ├── dfs.py
│   ├── greedy_best_first.py
│   ├── hill_climbing.py
│   ├── ids.py
│   ├── local_beam_search.py
│   ├── __pycache__
│   │   ├── a_star.cpython-310.pyc
│   │   ├── bfs.cpython-310.pyc
│   │   ├── dfs.cpython-310.pyc
│   │   ├── greedy_best_first.cpython-310.pyc
│   │   ├── hill_climbing.cpython-310.pyc
│   │   ├── ids.cpython-310.pyc
│   │   ├── local_beam_search.cpython-310.pyc
│   │   ├── simulated_annealing.cpython-310.pyc
│   │   └── ucs.cpython-310.pyc
│   ├── simulated_annealing.py
│   └── ucs.py
└── 7 directories, 31 files
```


Showcasing Results

```
$ python ./main.py
 0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30
0 #####
1 ##### A #####
2 #####
3 #####
4 #####
5 #####
6 # F ##### H B #
7 # ##### I #
8 # #####
9 # #####
10 # # G #####
11 # # #####
12 # # L ##### K #####
13 # E # ##### # C #
14 # # # #####
15 # # J #####
16 # #####
17 # M # ##### #
18 # ##### P #
19 # N # #####
20 # #####
21 # # # #####
22 # #####
23 ##### D #####
24 #####
{
  "gate_1": "A",
  "gate_2": "B",
  "gate_3": "C",
  "gate_4": "D",
  "gate_5": "E",
  "gate_6": "F",
  "admin_building": "G",
  "one_stop_shop": "H",
  "science_vally": "I",
  "academic_building": "J",
  "nano_building": "K",
  "helmy_building": "L",
  "service_building": "M",
  "work_shops": "N",
  "dorms": "P"
}
enter your current location: 
```

Showcasing Results

```
$ python ./main.py
 0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30
0 #####
1 ##### A #####
2 #####
3 #####
4 #####
5 #####
6 # F ##### H B #
7 # ##### I #
8 # #####
9 # #####
10 # # G #####
11 # # #####
12 # # L ##### K #####
13 # E # ##### # C #
14 # # # #####
15 # # J #####
16 # # #####
17 # M # ##### #
18 # ##### P #
19 # N # #####
20 # #####
21 # # # #####
22 # #####
23 ##### D #####
24 #####
{
  "gate_1": "A",
  "gate_2": "B",
  "gate_3": "C",
  "gate_4": "D",
  "gate_5": "E",
  "gate_6": "F",
  "admin_building": "G",
  "one_stop_shop": "H",
  "science_vally": "I",
  "academic_building": "J",
  "nano_building": "K",
  "helmy_building": "L",
  "service_building": "M",
  "work_shops": "N",
  "dorms": "P"
}
enter your current location: 
```

Showcasing Results

```
}  
enter your current location: gate_1  
enter your distantiation: service_building  
['gfs', 'a_star', 'bfs', 'dfs', 'ids', 'ucs', 'sim', 'hill', 'local']  
enter solving algorithm: ☐
```

Showcasing Results

```
enter your distantiation: service
['gfs', 'a_star', 'bfs', 'dfs', '']
enter solving algorithm: a_star
#####
#####*#####
###          |          ###
### ##### | #####
### ##### | #####
### ##### | #####
#F ##### | ##### H B##
## ##### | #####I ##
## ##### | ##### ##
## ##### | ##### ##
## # | --- #####
## # ##### | #####
## # L ##### | K #####
#E # ### ##### | ### ##### C#
#   ### ##### | ### ##### #
# # J | ##### #
# # ##### | ##### #
# + | # ##### | ### #
# ----- | ##### | P #
# N # ----- | ##### #
#           ##### #
#   ### ##### #
#   ##### #
#####D#####
#####
```

Uninformed Algorithms Showcase (BFS)

```
enter your distantiatio: service
['gfs', 'a_star', 'bfs', 'dfs', '']
enter solving algorithm: bfs
#####
#####*#####
###      |      ###
### #####| #####
### #####| #####
### #####| #####
#F #####| ##### H B##
## #####| #####I  ##
## #####| #####  ##
## #####--| ##### ##
## # -| G ##### ##
## #--| ##### ##### ##
## #| L ##### K ##### ##
#E #|### ##### ### ##### C#
# --|### ##### ### ##### #
# | # J ##### #
# | # ##### ##### #
# +| # ##### ### #
# ##### P #
# N # ##### #
# ##### #
# ### ##### #
# ##### #
#####D#####
#####
```

Uninformed Algorithms Showcase (IDS)

```
enter your distantiation: service
['gfs', 'a_star', 'bfs', 'dfs', '']
enter solving algorithm: ids
#####
#####*#####
##-----|#####
##|#####|#####
##|#####|#####
##|#####|#####
#-|#####|#####H B##
#|-#####|#####I##
#-|#####|#####
#|-#####|#####
##|-#G#####
#--|######
#|--#L#####K#####
#--|####|#####C#
#|--|###|#####
---|#J#####
|--|######
#+M#|#####
#|#####P#
#N#|#####
#|#####
#|#####
#|#####
#####D#####
#####
```

Uninformed Algorithms Showcase (UCS)

```
enter your distantiatio: service
['gfs', 'a_star', 'bfs', 'dfs', '
enter solving algorithm: ucs
#####
#####*#####
###          |          ###
### #####| #####
### #####| #####
### #####| #####
#F #####-| ##### H B##
## #####| #####I ##
## #####| #####
## #####---| #####
## # -| G #####
## # -| #####
## #| L ##### K #####
#E #|### ##### C#
# ---|### #####
# | # J #####
# | # #####
# +M # #####
# ##### P #
# N # #####
# #####
# #####
# #####
#####D#####
#####
```


Informed Algorithms Showcase (GFS)

```
enter your distantiation: service
['gfs', 'a_star', 'bfs', 'dfs', '']
enter solving algorithm: gfs
#####
#####*#####
###      |      ###
### #####| #####
### #####| #####
### #####| #####
#F #####| ##### H B##
## #####| #####I  ##
## #####| #####  ##
## #####| #####  ##
## #####| #####  ##
## #      |---  #####
## #      ####|  #####
## # L ####| K  #####
#E # ### ####| ### ##### C#
#   ### ####| ### ##### #
#   #      J |  ##### #
#   #      ####|  ##### #
# +| ----| ####|  ### #
# --|   | ####|  P  #
# N #   ----|  ##### #
#               ##### #
#   ###       ##### #
#   #####      #
#####D#####
#####
```

Informed Algorithms Showcase (A*)

```
enter your distantiatio: service
['gfs', 'a_star', 'bfs', 'dfs', '']
enter solving algorithm: a_star
#####
#####*#####
###      |      ###
### #####| #####
### #####| #####
### #####| #####
#F #####| ##### H B##
## #####| #####I  ##
## #####| #####  ##
## #####| #####  ##
## #####| #####  ##
## #      |---  #####
## #      |####| #####
## # L ####| K   #####
#E # ### ####| ### ##### C#
#   ### ####| ### ##### #
#   #      J | ##### #
#   #      ####| ##### #
# +| #      ####| ##### #
# - - - - -| ####| P   #
# N # - - - - -| ##### #
#               ##### #
#   ###          ##### #
#   #####      #
#####D#####
#####
```

Informed Algorithms Showcase (Hill Climbing)

```
['gfs', 'a_star', 'bfs', 'dfs',  
enter solving algorithm: hill  
#####  
#####*#####  
###      +---      ###  
### #####      #####  
### #####      #####  
### #####      #####  
#F #####      ##### H B##  
## #####      #####I  ##  
## #####      #####  ##  
## #####      #####  ##  
## #          G      #####  
## #          #####      ##  
## # L ##### K      #####  
#E # #####      ##### C#  
#   #####      ##### #  
#   #          J      #####  
#   #          #####      #  
# M #          #####      ##  
#   #####      P      #  
# N #          #####      #  
#           #####      #  
#   #####      #####      #  
#   #####      #####      #  
#####D#####  
#####  
(['down', 'right', 'left', 'left']
```

Informed Algorithms Showcase (Local Beam)

```
['gfs', 'a_star', 'bfs', 'dfs',  
enter solving algorithm: local  
#####  
#####A#####  
###                               ###  
### ##### ##### #####  
### ##### ##### #####  
### ##### ##### #####  
#F ##### ##### H *##  
## ##### #####I | ##  
## ##### ##### | ##  
## ##### ##### | ##  
## # G ##### | ##  
## # ##### ##### | ##  
## # L ##### K ##### | ##  
#E # ### ##### ### ##### | C#  
#   ### ##### ### ##### | #  
#   # J ##### | #  
#   # ##### ##### | #  
# M # ##### ## | #  
#   ##### +- | #  
# N # ##### #  
#   ##### #  
#   ### ##### #  
#   ##### #  
#####D#####  
#####  
(['down', 'down', 'down', 'down',
```

Informed Algorithms Showcase (Simulated Annealing)

```
['gfs', 'a_star', 'bfs', 'dfs', '
enter solving algorithm: sim
#####
#####A#####
##          ##
## #####   #####   #####
## #####   #####   #####
## #####   #####   #####
#F #####   #####   H   *##
## #####   #####I   |  ##
## #####   #####   |  ##
## #####   #####   |  ##
## #####   #####   |  ##
## #         G       #####| ##
## #         #####   #####| ##
## # L ##### K   #####| ##
#E # ## ##### ##   #####| C#
#   ## ##### ##   #####| - #
#   #         J       #####| #
#   #         #####   #####| #
# M #         #####   ###| #
#         #####   + --| #
# N #         #####   #
#         #####   #
#   ##         #####   #
#   #####   #
#####D#####
#####
(['down', 'down', 'down', 'down',
```