

# PROJECTE 1

## Navegador

SS: Sessió de Suport  
**Projecte 1 - Part 1**

---

Intel·ligència Artificial  
2023-2024  
Universitat Autònoma de Barcelona

## Objectiu: Proveu el vostre codi

Descarregueu el fitxer <testing file.py>

**cv.uab.cat**>>Pràctiques>>Projecte 1. Navegació

### Material del Projecte 1 [CAT]

En aquest apartat us deixem els enllaços als documents necessaris per aquest Proje

- <[Presentació P1pdf-CAT](#)>: Presentació del Projecte feta a la classe de Teoria.
- <[Practica1\\_1\\_CAT.pdf](#)>: Guia que us facilitarà la comprensió i resolució de la 1a P  
les funcions que heu de programar dins del fitxer SearchAlgorithm.py per la 1a Er
- <[PR1\\_Navigation.zip](#)>: Conjunt de fitxers que necessiteu per a la pràctica que in
  - Lyon\_smallCity: Mapa de Lyon per començar a treballar. Només serveix a l'inici  
scripts que actuen com a tests. /
  - Lyon\_bigCity: Mapa de Lyon complet amb el qual haurà de funcionar la pràctica
  - CityInformation: fitxers que descriuen una ciutat
  - Code: Conjunt de fitxers python que us entreguem. Consulteu Practica1\_1.pdf
- <[SO\\_Part 1\\_Presentacio](#)>: Presentació de la Sessió d'Orientació de la Part 1.
- <[SS\\_Part 1\\_Presentacio](#)>: Presentació de la Sessió de Suport de la Part 1.
- <[testing file.py](#)>: Fitxer per provar les funcions de test.

## Què hi ha en aquest fitxer?

<testing file.py>

```
from SearchAlgorithm import *
from SubwayMap import *
from utils import *

if __name__=="__main__":
    ROOT_FOLDER = '../CityInformation/Lyon_smallCity/'
    map = read_station_information(os.path.join(ROOT_FOLDER, 'Stations.txt'))
    connections = read_cost_table(os.path.join(ROOT_FOLDER, 'Time.txt'))
    map.add_connection(connections)

    infoVelocity_clean = read_information(os.path.join(ROOT_FOLDER, 'InfoVelocity.txt'))
    map.add_velocity(infoVelocity_clean)

    ###BELOW HERE YOU CAN CALL ANY FUNCTION THAT YOU HAVE PROGRAMED TO ANSWER THE QUESTIONS FOR THE TEST###

    #example
    example_path = expand(Path([5]), map)
    print_list_of_path([example_path])
```

```
from SearchAlgorithm import *
from SubwayMap import *
from utils import *
```

*Aquí s'importen tots els fitxers que contenen les funcions que necessita el vostre codi*

```
if __name__=="__main__":
    ROOT_FOLDER = '../CityInformation/Lyon_smallCity/'
    map = read_station_information(os.path.join(ROOT_FOLDER, 'Stations.txt'))
    connections = read_cost_table(os.path.join(ROOT_FOLDER, 'Time.txt'))
    map.add_connection(connections)

    infoVelocity_clean = read_information(os.path.join(ROOT_FOLDER, 'InfoVelocity.txt'))
    map.add_velocity(infoVelocity_clean)
```

###BELOW HERE YOU CAN CALL ANY FUNCTION THAT YOU HAVE PROGRAMED TO ANSWER THE QUEST  
FOR THE TEST###

```
#example
example_path = expand(Path([5]), map)
print_list_of_path([example_path])
```

```
from SearchAlgorithm import *
from SubwayMap import *
from utils import *
```

*Aquí establim els paths i carreguem les dades del mapa ...*

```
if __name__=="__main__":
    ROOT_FOLDER = '../CityInformation/Lyon_smallCity/'
    map = read_station_information(os.path.join(ROOT_FOLDER, 'Stations.txt'))
    connections = read_cost_table(os.path.join(ROOT_FOLDER, 'Time.txt'))
    map.add_connection(connections)

    infoVelocity_clean = read_information(os.path.join(ROOT_FOLDER, 'InfoVelocity.txt'))
    map.add_velocity(infoVelocity_clean)
```

###BELOW HERE YOU CAN CALL ANY FUNCTION THAT YOU HAVE PROGRAMED TO ANSWER THE QUEST  
FOR THE TEST###

```
#example
example_path = expand(Path([5]), map)
print_list_of_path([example_path])
```

```
from SearchAlgorithm import *
from SubwayMap import *
from utils import *
```

```
if __name__=="__main__":
    ROOT_FOLDER = '../CityInformation/Lyon_smallCity/'
    map = read_station_information(os.path.join(ROOT_FOLDER, 'Stations.txt'))
    connections = read_cost_table(os.path.join(ROOT_FOLDER, 'Time.txt'))
    map.add_connection(connections)
```

```
infoVelocity_clean = read_information(os.path.join(ROOT_FOLDER, 'InfoVelocity.txt'))
map.add_velocity(infoVelocity_clean)
```

*Aquí carreguem les dades sobre les velocitats de les línies*

###BELOW HERE YOU CAN CALL ANY FUNCTION THAT YOU HAVE PROGRAMED TO ANSWER THE QUEST  
FOR THE TEST###

```
#example
example_path = expand(Path([5]), map)
print_list_of_path([example_path])
```

```
from SearchAlgorithm import *
from SubwayMap import *
from utils import *

if __name__=="__main__":
    ROOT_FOLDER = '../CityInformation/Lyon_smallCity/'
    map = read_station_information(os.path.join(ROOT_FOLDER, 'Stations.txt'))
    connections = read_cost_table(os.path.join(ROOT_FOLDER, 'Time.txt'))
    map.add_connection(connections)

    infoVelocity_clean = read_information(os.path.join(ROOT_FOLDER, 'InfoVelocity.txt'))
    map.add_velocity(infoVelocity_clean)
```

###BELOW HERE YOU CAN CALL ANY FUNCTION THAT YOU HAVE PROGRAMED TO ANSWER THE QUESTIONS FOR THE TEST###

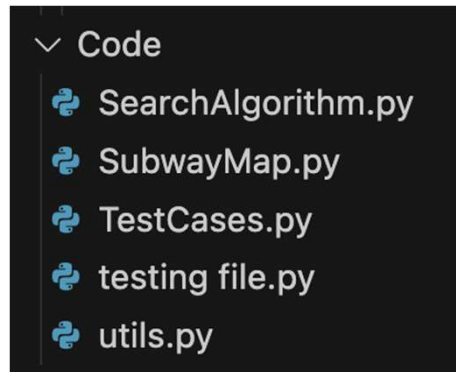
*Aquí és on copiareu les línies de codi que haureu d'executar per contestar les preguntes del qüestionari*

```
#example
example_path = expand(Path([5]), map)
print_list_of_path([example_path])
```

## Anem al qüestionari wooclap

El qüestionari que fareu a continuació us proposarà un seguit de proves per a les vostres funcions de **SearchAlgorithm.py**,

Obriu ara el fitxer '**testing file.py**' i ubiqueu-lo al mateix directori que la resta de fitxers de codi.



Per contestar el qüestionari haureu de provar cada un dels casos plantejats,

Si no contesteu bé a les preguntes, **haureu d'esbrinar què falla al vostre codi ...**



## Responen ràpid!

**Atenció! Només 1 minut i mig per pregunta.**

Per veure el resultat, feu ús de funcions *print* com les que teniu al fitxer **utils.py**.

```
utils.py  ×
Code > utils.py > ...
46
47 def print_list_of_path(path_list):
48     for p in path_list:
49         print("Route: {}".format(p.route))
50
51
52 def print_list_of_path_with_cost(path_list):
53     for p in path_list:
54         print("Route: {}, \t Cost: {}".format(p.route, p.g))
55
```












Ara sí,  
ja anem al qüestionari

SS-CAT-Project 1-Part 1 ✕ ✓ Participate at: [app.wooclap.com/NCLZGO](https://app.wooclap.com/NCLZGO) 📄 ⚙️ Set

[Votes](#) [Messages](#) [Participant pace](#) 👤 JCM 🔄 Share ➕ Add present

💡 Examples gallery 📁 Import questions ➕ New quest

ℹ️ How to participate? Display

	1. Quines d'aquestes funcions ja has codificat i testejat?	Edit	Display
	2. Corre la funció <code>expand(Path([2]), map)</code> . Quina de les següents opcions NO és un output de la funció?	Edit	Display
	3. Corre la funció <code>expand(Path([4, 5]), map)</code> . Quina opció de les següents és un dels outputs de la funció?	Edit	Display
	4. Què succeeix si apliques la funció <code>remove_cycles</code> al resultat de la següent funció <code>expand(Path([4, 5, 6])...</code>	Edit	Display
	5. Corre la funció <code>remove_cycles([Path([7,7]))]</code> . Quina és la sortida?	Edit	Display
	6. Quina és la sortida de <code>depth_first_search(3, 10, map)</code> ?	Edit	Display
	7. Quina és la sortida de <code>depth_first_search( 10 , 10, map )</code> ?	Edit	Display
	8. Ets a l'estació 1 i vols anar a l'estació 10. Quin mètode de cerca usaries per trobar el camí més curt?	Edit	Display
	9. Què retorna <code>breadth_first_search( 10, 10000, map)</code> ?	Edit	Display
	10. Corre la funció <code>distance_to_stations( [0,0], map)</code> i determina l'estació més propera.	Edit	Display
	11. Quines funcions necessites corregir i tornar a testear de nou?	Edit	Display

## Recordeu: Com heu de fer l'entrega ...

Per a l'avaluació d'aquesta primera part de la pràctica haureu de pujar al Campus Virtual el vostre fitxer **SearchAlgorithm.py** que ha de contenir el vostre **NIU** a la variable autors (a l'inici de l'arxiu).

L'entrega s'ha de fer abans del dia **03/03/2023 a les 23:55.**

**ATENCIÓ!** és important que tingueu en compte els següents punts:

1. La **correcció** del codi es fa de manera **automàtica**, per tant, assegureu-vos de penjar els arxius amb la nomenclatura i format correctes. Si no ho poseu bé la nota serà un 0.
2. El codi està sotmès a **detecció automàtica** de plagis durant la correcció.
3. Qualsevol part del codi que no estigui dins de les funcions de l'arxiu SearchAlgorithm.py **no** podrà ser **avaluada**, per tant, no modifiqueu res fora d'aquest arxiu.
4. Per evitar que el codi entri en bucles infinits hi ha un **límit de temps** per a cada exercici, per tant si les vostres funcions triguen massa les considerarà incorrectes.