Assignment 2 Uninformed and informed Search

Aims:

To perform an uninformed and an informed search for a given problem in a search space organized as a tree.

Task:

Specify, design and deploy an application in python that solve your assigned problem using the specified search methods. The applications should follow the following conditions:

1. It must have a nice architecture (for example the following UML diagram - you can add functions and classes as need it)

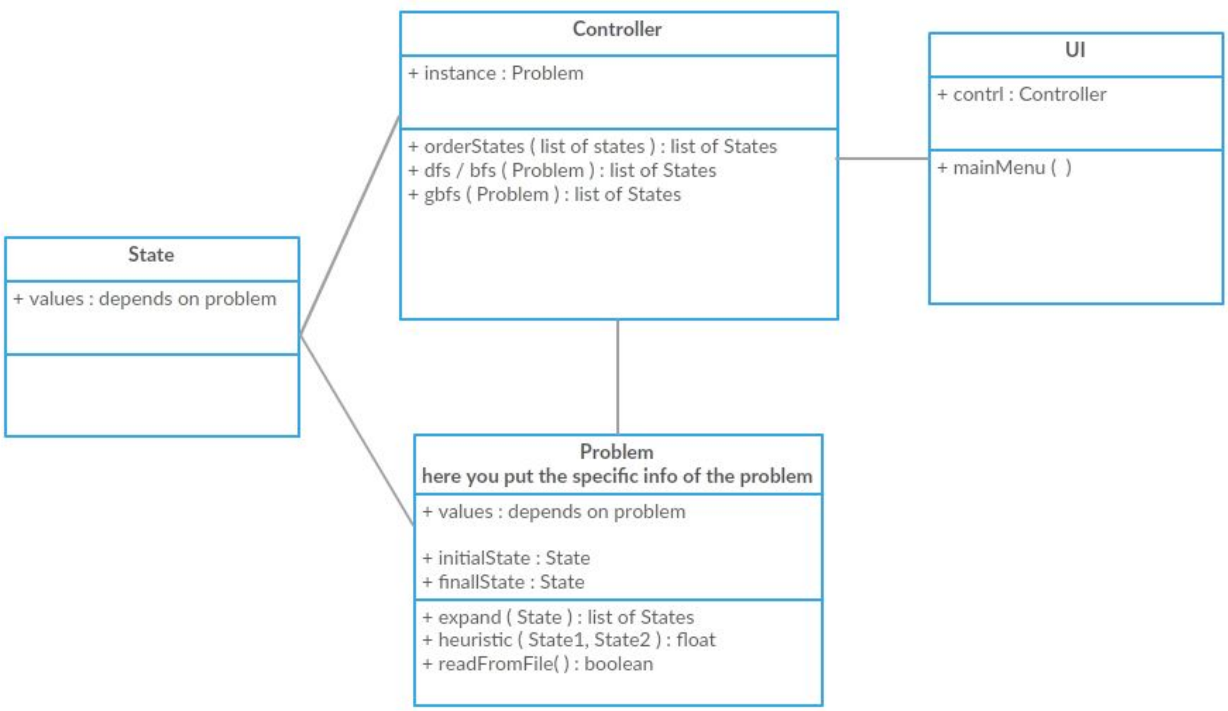


Figure 1: The UML diagram

2. the input data will be in a text file
3. the user can choose in a text menu the method that will be used to solve the problem

Each problem must be solved with both two methods!!!
AND NOT WITH OTHER ONES!!!

Points:

- 40 points / method.
- 20 points for the architecture and for the quality of your application.

Time:

1 h 30 min (the deadline is at the end of the second lab.)

General hints:

- Determine the search tree according to your problem! Will help you A LOT!
- Do not implement functions that you will NEVER use in your application!
- Try to keep the solution simple - these are not difficult problems.
- Ask if you don't know how to solve it! Time is important!
- Do NOT solve the problems with other methods. You will not be granted points if you do this.

Problems:

1. The *n*-queen problem – solving techniques: BFS, GBFS

Consider a chess board with *n* x *m* squares. Determine the maximum number of queens that can be placed on the board in such way that they can't attack each other. In **Figure 2** there are two examples (correct and incorrect) for *n = 5* and *m = 6*.

	R				
			R		
					R
		R			
				R	

Figure 2: a) a valid configuration

	R				
			R		
					R
				R	
		R			

b) invalid configuration

2. Sudoku game – solving techniques: BFS, GBFS

Consider a Sudoku game - a logic puzzle represented on a *n* x *n* board; some squares contain already a number, others must be completed with other numbers from {*1,2,…,n*} in such a way that each line, column and square with the edge equal with √n must contain only different numbers. Determine one correct solution for the puzzle.

3			2
	1	4	
1	2		4
	3	2	1

	2		6		8			5
5	8				9	7		
		7		4			2	8
3	7		4		1	5		
6				8				5
		8			2		1	3
8		6		2		1		
		9	8				3	6
7			3		6		9	

Figure 3: a) Sudoku game with 4x4 squares; b) Sudoku game with 9x9 squares

3. Cryptarithmetic game – solving techniques: DFS, GBFS

Implement an algorithm that solves a crypt-arithmetic problem as the ones presented in **Figure 4** knowing that:

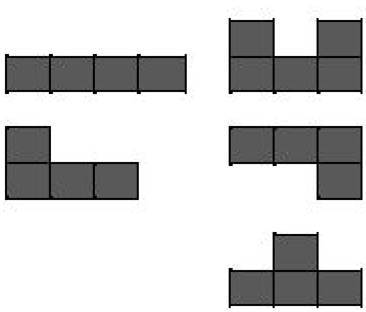
- Each letter represent a hexadecimal cipher;
- The result of the arithmetic operation must be correct when the letters are replaced by numbers;
- The numbers can not start with 0;
- Every problem can have only one solution.

```
 SEND+        TAKE+        EAT+       NEVER –
 MORE=           A        THAT=        DRIVE=
------       CAKE=       -----       ------
MONEY       ------       APPLE         RIDE
             KATE
```

Figure 4: Cryptarithmetic problems

4. Geometric forms – solving techniques: DFS, GBFS

Consider the geometric forms from **Figure 5**. Determine an arrangement for this forms on a square board of 5x6 in such a way that the board will be uniform covered and the forms will not overlap.



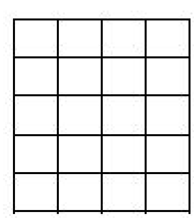


Figure 5: a) the geometric forms. b) the game board.

5. The problem of sorting objects – solving techniques: DFS, GBFS

Ordinate a list of objects (for example a list of integers) using an interchange operator between two objects from the list.

6. The travel salesman's problem – solving techniques: BFS, GBFS

For a given map with cities from Europe and distances between them (see **Figure 6**), find the shortest way from Bucharest to Paris knowing that the direct distances between the cities are in **Table 1** and **Table 2**.

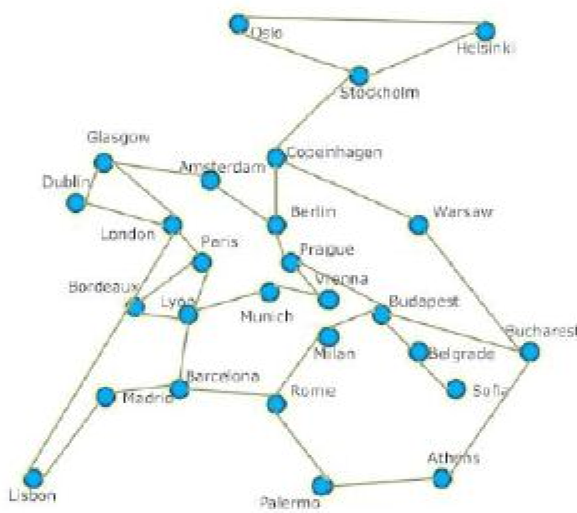


Figure 6: The map of european cities.

Olso - Helsinki:	970	Rome: Milan:	681	Madrid - Barcelona:	628
Helsinki - Stockholm:	400	Milan - Budapest:	789	Madrid - Lisbon:	638
Oslo - Stockholm:	570	Vienna - Budapest:	217	Lisbon - London:	2210
Stockholm - Copenhagen:	522	Vienna - Munich:	458	Barcelona - Lyon:	644
Copenhagen - Warsaw:	668	Prague - Vienna:	312	Paris - London:	414
Warsaw - Bucharest:	946	Prague - Berlin:	354	London - Dublin:	463
Bucharest - Athens:	1300	Berlin - Copenhagen:	743	London - Glasgow:	667
Budapest - Bucharest:	900	Berlin - Amsterdam:	648	Glasgow - Amsterdam:	711
Budapest - Belgrade:	316	Munich - Lyon:	753	Budapest - Prague:	443
Belgrade - Sofia:	330	Lyon - Paris:	481	Barcelona - Rome:	1471
Rome - Palermo:	1043	Lyon - Bordeaux:	542	Paris - Bordeaux:	579
Palermo - Athens:	907			Glasgow - Dublin:	306

Table 1: The distances between the cities from map shown in Figure 6.

Amsterdam	2280	Copenhagen	2250	Madrid	3300	Rome	1140
Athens	1300	Dublin	2530	Milan	1750	Sofia	390
Barcelona	2670	Glasgow	2470	Munich	1600	Stockholm	2890
Belgrade	630	Helsinki	2820	Oslo	2870	Vienna	1150
Berlin	1800	Lisbon	3950	Palermo	1280	Warsaw	946
Bordeaux	2100	London	2590	Paris	2970		
Budapest	900	Lyon	1660	Prague	1490		

Table 2: The direct distances between Bucharest and the other cities from Figure 6.

7. The missionaries and cannibals problem – solving techniques: DFS, GBFS

Three missionaries and three cannibals are on the left side of a river. They want to cross over the right side with the help of a boat that can only transport two persons per trip. The number of the cannibals can not be higher than the number of missionaries on one side because the missionaries will be eaten by the cannibals. Find a solution to cross all of them on the other side.

8. The crossing river problem – solving techniques: BFS, GBFS

On one side of a river there are an adult, two children and a tiny little boat. The people want to cross the river by boat. The tiny little boat can take either two children, either one child, either one adult. Determine a sequence of moves to cross the river.

9. The sliding puzzle problem – solving techniques: BFS, GBFS

For a given puzzle of *n* x *n* squares with numbers from 1 to (*n* x *n-1*) (one square is empty) in an initial configuration, find a sequence of movements for the numbers in order to reach a final given configuration, knowing that a number can move (horizontally or vertically) on an adjacent empty square. In **Figure 7** are presented two examples of puzzles (with the initial and final configuration).

	3
1	2

OI

2	1
3	

OF

4	7	3
1	5	8
6	2	

OI

	1	2
3	4	5
6	7	8

OF

Figure 7: a) sliding puzzle with n=2 b) sliding puzzle with n=3

(OI – initial order, OF – final order)

10. The farmer`s problem – solving techniques: BFS, GBFS

A farmer, a goat, a wolf and a cabbage were on one side of a river and they want to cross on the other side using a boat. Find a way to cross it knowing that:

- The boat can fit only two “passengers”;
- The wolf and the goat can NOT stay alone on the same side of the river;
- The cabbage and the goat can NOT stay on the same side of the river.

11. Jumping over cubes problem – solving techniques: DFS, GBFS

Identify a descending path from the top of a pyramid built from $1+2^2+3^2+\ldots+n^2$ cubes of same dimension, but with different associated scores, in such a way that the total score cumulated on the path would be maxim. The pyramid is build it such that one cube is placed over other 4 cubes – one quarter of the cube is over one (from the 4) below it. The descend from a level to another implies a jump from a cube c_i from level i to a cube c_{i+1} from level $i+1$ (the top cube is on level 1, the bottom cubes are on level n), the cube c_i being over the cube c_{i+1} (see **Figure 8**).

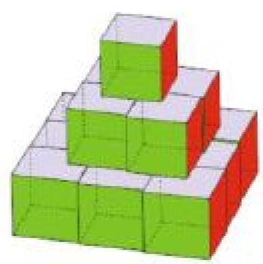


Figure 8: An example of pyramid.

12. Triangle problem – solving technique: DFS, GBFS

Identify a path from the top to the bottom in a triangle composed from $1+2+3+\ldots+n$ squares of same dimension. Each square has different associated scores. The desired path has the maximum cumulated score.

The triangle has several levels, each level holding a number of squares. The top square is placed on the level 1, the bottom are on level n.

Traversing implies going down or moving on the same level. Descending from a level to another is a jump from the square $p_{i,j}$ from level i to a square $p_{i+1,j}$ from level $i+1$, situated on the same column j (see **Figure 9**). Moving (left or right) implies making steps form a square $p_{i,j}$ from the level i, column j on an adjacent square from the same level i, but on column $j-1$ or $j+1$.

				9				
			5	7	2			
		5	8	2	4	1		
	6	3	1	9	4	3	2	
4	5	7	2	3	4	5	1	9

Figure 9: An example of a triangle of squares.

13. Labyrinth problem – solving technique: DFS, GBFS

To make a lighting system, the paths in a labyrinth (see **Figure 10**) were divided on squares. Each square has an associated cost. Determine all the possibilities to exit the labyrinth and the cheapest path out starting from an interior given position.

	3			1			
	1		6	9	4	3	
	9	7	7			1	5
4	7		3	6	2	7	
	5		5		8		
		4		5	2		
2	8	3			7	8	1
		8		4			

Figure 10: Labyrinth example.

14. Phone line problem – solving technique: DFS, GBFS

In an office building with many levels is set up a phone line cable. For some reasons is not possible to place the line in every office (there is not sufficient cable) identify an optimal way to distribute the cable in the office so a maximum number of employees can talk to the phone. We also know that:

- At each level is the same number of offices;
- In every office there is a certain know number of employees;
- The cable starts from the top left office and must end on the bottom right office;
- The cable can go from an office to another only if the two are adjacent (on horizontal or vertical).

15. The clocks problem – solving technique: DFS, GBFS

In an airport on a wall there are 9 clocks that show the hours for different cities from the globe. The clocks are placed like in a matrix with 3 lines and 3 columns. Due to a power failure, the clocks were disrupted and needs to be adjusted again. For setup there are 2 buttons that allow the following moves:

- Turning the first one up make all the clocks from a line increase the hour with one unit. Turning it down decreases an hour to all the clocks from one line;
- Turning the second up increases with one hour the time to all the clocks from one column, respectively turning it down decreases with one hour the time from all the clocks on a column.

Please set up all clocks at the correct time using as little as possible the two buttons.

16. Mouse problem – solving technique: DFS, GBFS

In **Figure 11** is a labyrinth of *5x5* squares in which can be mouse traps and a cheese “treasure”. In a corner is a hungry little mouse that smells the cheese but doesn’t know where it is. Help the mouse to get to the “treasure” without being caught in the traps knowing that it can move either on horizontal or vertically.

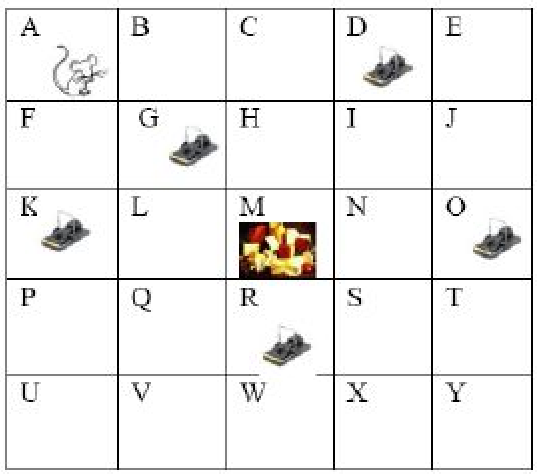


Figure 11: Mouse “treasure” hunt.

17. The construct site problem – solving technique: DFS, GBFS

A construction company has 4 excavators (A-D) and they have to use them to finalize 4 projects (W, X, Y, Z). The excavators are not efficient regarding the petrol consumption so the company has to minimize the costs. Help them to make a schedule and to assign the excavators to the tasks knowing that one excavator can work one project of a time and it has a certain costs given in **Figure 12**.

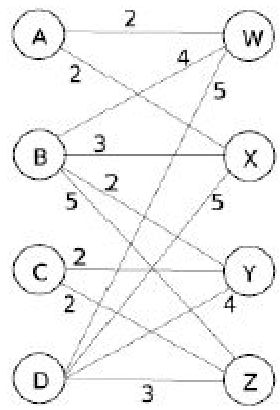


Figure 12: Associated costs for the excavators and the projects.

18. Color the map problem – solving technique: BFS, GBFS

There is a map with several countries (we know what country is neighbor to another) and 3 colors. Find a way to color the map in such way that there are no 2 adjacent countries with the same color.

19. Dividing the presents problem – solving technique: DFS, GBFS

Two brothers receive in a box several bags with candies, (in bags the number of candies differs) and they want to split them in a brotherhood manner so each receives the same number of candies as the other. Help them identify a method to spit the bags between them.

20. Date game – solving technique: BFS, GBFS

Three boys and three girls using seven chairs play the following game: The kinds sit on 6 of the 7 available chairs, initially all boys are next to each other, after is an empty chair, and after all the girls (one next to another) – see also **Figure 13**. The game implies making 2 kinds of moves:

- One person can move on an empty chair next to him, with a cost of one unit;
- A person can move on an empty chair over one or two persons, with a cost equal with the number of persons that was jumped over plus one.

Identify an optimal way to move the persons so they form pairs boy-girl on adjacent chairs - see **Figure 14**.

Figure 13: Date game – initial configuration.

Figure 14: Date game – final configuration.

21. Frobenius' problem (the coins exchange problem) – solving technique: BFS, GBFS

There are n coin types of values $m_1, m_2, ..., m_n$ (integers) such that $m_i>1$, for $i =1,2,..,n$ and $cmmdc(m_1,m_2,...,m_n) = 1$. Find the highest integer *NF* (Frobenius' number) that can't be express as a linear combination of the *n* coins ($NF = a_1m_1+a_2m_2+...a_nm_n$).