

# Tema 1 – Inteligenta Artificiala

## Cuprins:

1. Detalii Problema	pag 1
2. Executare	pag 2
3. Memorarea Starii	pag 3
4. Generarea Succesorilor	pag 4
5. Euristici	pag 4
6. Exemple	pag 5
7. Validari si Optimizari	pag 6
8. Valori	pag 6
8.1) UCS.....	pag 6
8.2) A* Neoptimizat.....	pag 7
8.3) A* Optimizat.....	pag 7
8.4) IDA*.....	pag 7
8.5) Alte exemple de afisari.....	pag 8

## 1 Detalii Problema

Am avut de rezolvat “Problema vaselor cu apa”, care consta in:

### Context:

Consideram ca avem niste vase cu apa colorata. Despre fiecare vas stim capacitatea maxima si cat lichid contine. Pot exista si vase vide. De asemenea pentru combinatia a doua culori de lichide stim ce culoare rezulta din combinatia lor. Pentru combinatiile de culori neprecizate, inseamna ca nu ne intereseaza rezultatul si desi

le putem amesteca (uneori e nevoie sa depozitam apa intr-un vas, ca sa facem loc pentru alte mutari) culoarea rezultata nu va aparea in starea solutie niciodata (puteti considera un identificator special pentru acea culoare, de exemplu "nedefinit"). Evident, apa cu culoare nedefinita nu poate fi folosita pentru a obtine alte culori (apa cu culoare nedefinita, amestecata cu orice rezulta in culoare nedefinita).

### **Mutări:**

Mutările se fac ținând cont de următoarele reguli:

- Lichidul dintr-un vas nu poate fi varsat decat in alt vas (nu dorim sa pierdem din lichid; nu se varsa in exterior).
- Indiferent de cantitatea de lichid turnată și cea existentă in vas, culoarea rezultată în vasul în care s-a turnat apa e fie e culoarea indicată în fișierul de intrare pentru combinarea celor două culori, fie nedefinită dacă nu se specifică în fișierul de intrare rezultatul unei astfel de combinări. Apa de culoare nedefinită, turnată peste orice altă culoare, va transforma apa din vasul în care se toarnă în apă de culoare nedefinită
- Apa se poate turna dintr-un vas în altul doar în două moduri: fie se toarnă apă până se golește vasul din care turnăm, fie se umple vasul în care turnăm. Nu se pot turna cantități intermediare.

### **Starea finala (scopul):**

Considerăm că ajungem în starea finală când obținem cantități fixe (cerute în fișierul de intrare) de apă de o anumită culoare.

Codul a fost implementat in python si are o structura asemanatoare cu cea din laborator.

## **2 Executare**

Executia programului se realizeaza din fisierul "main.py" (in terminal: python3 main.py), urmand a fi completate informatiile cerute de catre interpretor.

Exemplu de executie:

```
PS D:\IA\Tema1> python3 main.py
In ce fisier putem gasi date de intrare?
$ Samples
Alegeti un folder in care sa salvam solutiile obtinute!
$ Outputs
Cate solutii sa afiseze algoritmul (A*/IDA* obtin doar prima solutie)?
$ 2
Acestea sunt estimarile valabile:
  1. estimarea penala
  2. estimare admisibila 1
  3. estimare admisibila 2
  4. estimare inadmisibila
Care estimare doriti sa o folosim in algoritm?
$ 3
Algoritmii de cautare valabili:
  1. UCS
  2. A_Star
  3. A_Star_Optimizat
  4. IDA_Star
Care cautare doriti sa o folosim?
$ 1
Ce timeout sa aiba algoritm-ul (in secunde)?
$ 10
Procesare date input din fisierul "date_input.txt" ... Am gasit sol
Am gasit sol
Procesare date input din fisierul "file1.txt" ... Procesare date input din fisierul "file2.txt" ... Am gasit sol
Procesare date input din fisierul "file3.txt" ... Am gasit sol
Am gasit sol
PS D:\IA\Tema1> 
```

### 3 Memorarea Starii

In starea mea memorez urmatoarele informatii:

- a) Capacitatea maxima care poate fi retinuta intr-un vas

- b) Capacitatea aflata actual in vasul respective
- c) Culoarea apei din vasul respective
- d) Starea precedenta
- e) Distanta de la starea initiala la cea actuala
- f) Combinatiile posibile de culori ( Ex: { a, b, c } = a combinate cu b -> c )
- g) Daca este sau nu nod scop

## 4 Generarea Succesorilor

Pentru a genera succesorii, folosesc urmasorii pasi:

- Verific daca vasul din care vreau sa torn are apa in el
- Verific daca vasul in care vreau sa torn este deja plin
- Daca pot turna dintr-un vas in altul tin minte cantitatea de apa turnata
- Modific astfel, in functie de punctul anterior, informatiile din primul vas, respectiv al doilea vas
- Realizez noul succesor
- Verific daca acesta a fost gasit intr-o etapa anterioara
- Daca nu a fost gasit intr-o etapa anterioara il adaug la lista mea de succesor

## 5 Euristici

Programul scris contine 4 euristici, respectiv:

- 1) **Euristica banala:** Fiecare stare are distanta pana la o stare finala egala cu 0. Inevitabil, costul estimate va fi mai mic decat costul adevarat.
- 2) **Euristica admisibila 1:** Returnez numarul de culori lipsa pana la atingerea satisfacerii nodului scop. Euristica este corecta, deoarece numarul

de culori lipsa va fi mereu minim (intotdeauna  $\leq$  decat nr de culori din nodul scop)

- 3) **Euristica admisibila 2:** Returnez numarul de vase care mai trebuiesc pana cand vor fi satisfacute atat culoarea din nodul scop, cat si cantitatea de apa de culoarea respective ceruta. De asemenea, aceasta euristica este corecta.
- 4) **Euristica neadmisibila:** Daca nodul este nod final returnez 10, iar in caz contrar returnez o valoare inmultita cu Euristica Admisibila 2 din care scad produsul dintre o valoare si Estimarea Admisibila 1. In mod evident, euristica va fi gresita.

## 6 Exemple

In fisierul Samples se vor regasi fisiere pentru date de intrare astfel incat:

- a. Un fisier de input care nu are solutii: file1.txt
- b. Un fisier de input care da o stare initiala care este si finala: file2.txt
- c. Un fisier de input care nu blocheaza pe niciun algoritm si sa aiba ca solutii drumuri lungime micuta, sa zicem de lungime maxim 20:  
date\_input.txt
- d. Un fisier sa dea drumul de cost minim pentru euristicile admisibile si sa nu gaseasca solutie pentru euristica neadmisibila: file3.txt

## 7 Validari si optimizari

Datele memorate din fisierul de intrare vor trece prin verificari, care asigura daca acestea sunt valide sau nu.

## 8 Valori

### 8.1 Uniform Cost Search

```
Trebuie sa afisez nod
distanta: 0
vase: [(5, 4, 'rosu'), (2, 2, 'galben'), (3, 0, 'nedefinit'), (7, 3, 'albastru'), (1, 0, 'nedefinit'), (4, 3, 'rosu')]
distanta: 1
vase: [(5, 1, 'rosu'), (2, 2, 'galben'), (3, 3, 'rosu'), (7, 3, 'albastru'), (1, 0, 'nedefinit'), (4, 3, 'rosu')]
distanta: 2
vase: [(5, 1, 'rosu'), (2, 1, 'galben'), (3, 3, 'rosu'), (7, 3, 'albastru'), (1, 1, 'galben'), (4, 3, 'rosu')]
distanta: 3
vase: [(5, 1, 'rosu'), (2, 2, 'verde'), (3, 3, 'rosu'), (7, 2, 'albastru'), (1, 1, 'galben'), (4, 3, 'rosu')]
distanta: 4
vase: [(5, 0, 'rosu'), (2, 2, 'verde'), (3, 3, 'rosu'), (7, 3, 'mov'), (1, 1, 'galben'), (4, 3, 'rosu')]
Timpul gasirii solutie a fost de: 2731 ms!
```

```
Trebuie sa afisez nod
distanta: 0
vase: [(5, 4, 'rosu'), (2, 2, 'galben'), (3, 0, 'nedefinit'), (7, 3, 'albastru'), (1, 0, 'nedefinit'), (4, 3, 'rosu')]
distanta: 1
vase: [(5, 1, 'rosu'), (2, 2, 'galben'), (3, 3, 'rosu'), (7, 3, 'albastru'), (1, 0, 'nedefinit'), (4, 3, 'rosu')]
distanta: 2
vase: [(5, 1, 'rosu'), (2, 1, 'galben'), (3, 3, 'rosu'), (7, 3, 'albastru'), (1, 1, 'galben'), (4, 3, 'rosu')]
distanta: 3
vase: [(5, 1, 'rosu'), (2, 2, 'verde'), (3, 3, 'rosu'), (7, 2, 'albastru'), (1, 1, 'galben'), (4, 3, 'rosu')]
distanta: 4
vase: [(5, 3, 'mov'), (2, 2, 'verde'), (3, 3, 'rosu'), (7, 0, 'albastru'), (1, 1, 'galben'), (4, 3, 'rosu')]
Timpul gasirii solutie a fost de: 5465 ms!
Detalii UCS
```

## 8.2 Slow A\*

```
Trebuie sa afisez nod
distanta: 0
vase: [(5, 4, 'rosu'), (2, 2, 'galben'), (3, 0, 'nedefinit'), (7, 3, 'albastru'), (1, 0, 'nedefinit'), (4, 3, 'rosu')]
distanta: 1
vase: [(5, 5, 'nedefinit'), (2, 1, 'galben'), (3, 0, 'nedefinit'), (7, 3, 'albastru'), (1, 0, 'nedefinit'), (4, 3, 'rosu')]
distanta: 2
vase: [(5, 5, 'nedefinit'), (2, 2, 'verde'), (3, 0, 'nedefinit'), (7, 2, 'albastru'), (1, 0, 'nedefinit'), (4, 3, 'rosu')]
distanta: 3
vase: [(5, 5, 'nedefinit'), (2, 2, 'verde'), (3, 2, 'albastru'), (7, 0, 'albastru'), (1, 0, 'nedefinit'), (4, 3, 'rosu')]
distanta: 4
vase: [(5, 5, 'nedefinit'), (2, 2, 'verde'), (3, 3, 'mov'), (7, 0, 'albastru'), (1, 0, 'nedefinit'), (4, 2, 'rosu')]
Timpul gasirii solutie a fost de: 4223 ms!
```

## 8.3 A\* Optimizat

```
Trebuie sa afisez nod
distanta: 0
vase: [(5, 4, 'rosu'), (2, 2, 'galben'), (3, 0, 'nedefinit'), (7, 3, 'albastru'), (1, 0, 'nedefinit'), (4, 3, 'rosu')]
distanta: 1
vase: [(5, 5, 'nedefinit'), (2, 1, 'galben'), (3, 0, 'nedefinit'), (7, 3, 'albastru'), (1, 0, 'nedefinit'), (4, 3, 'rosu')]
distanta: 2
vase: [(5, 5, 'nedefinit'), (2, 2, 'verde'), (3, 0, 'nedefinit'), (7, 2, 'albastru'), (1, 0, 'nedefinit'), (4, 3, 'rosu')]
distanta: 3
vase: [(5, 5, 'nedefinit'), (2, 2, 'verde'), (3, 2, 'albastru'), (7, 0, 'albastru'), (1, 0, 'nedefinit'), (4, 3, 'rosu')]
distanta: 4
vase: [(5, 5, 'nedefinit'), (2, 2, 'verde'), (3, 3, 'mov'), (7, 0, 'albastru'), (1, 0, 'nedefinit'), (4, 2, 'rosu')]
Timpul gasirii solutie a fost de: 913 ms!
```

## 8.4 IDA\*

```
Trebuie sa afisez nod
distanta: 0
vase: [(5, 4, 'rosu'), (2, 2, 'galben'), (3, 0, 'nedefinit'), (7, 3, 'albastru'), (1, 0, 'nedefinit'), (4, 3, 'rosu')]
distanta: 1
vase: [(5, 1, 'rosu'), (2, 2, 'galben'), (3, 3, 'rosu'), (7, 3, 'albastru'), (1, 0, 'nedefinit'), (4, 3, 'rosu')]
distanta: 2
vase: [(5, 1, 'rosu'), (2, 1, 'galben'), (3, 3, 'rosu'), (7, 3, 'albastru'), (1, 1, 'galben'), (4, 3, 'rosu')]
distanta: 3
vase: [(5, 1, 'rosu'), (2, 2, 'verde'), (3, 3, 'rosu'), (7, 2, 'albastru'), (1, 1, 'galben'), (4, 3, 'rosu')]
distanta: 4
vase: [(5, 0, 'rosu'), (2, 2, 'verde'), (3, 3, 'rosu'), (7, 3, 'mov'), (1, 1, 'galben'), (4, 3, 'rosu')]
Timpul gasirii solutie a fost de: 80 ms!
```

## 8.5 Exemple de output pentru anumite fisiere de intrare

```
Trebuie sa afisez nod  
distanta: 0  
vase: [(5, 2, 'rosu'), (3, 1, 'albastru')]  
Starea initiala este egala cu starea finala.  
Timpul gasirii solutie a fost de: 0 ms!  
Detalii UCS
```

```
Nu au fost gasite solutii!  
Detalii UCS
```