

Artificial intelligence - Project 3
- Limbaje de planificare -

Pop Ruxandra Maria, Zelenszky Bianca

14/01/2022

1 Introducere

Pentru realizarea acestui proiect au fost nevoie de cunoștințe legate de planificarea unei probleme și găsirea pașilor care duc la un scop bine stabilit. Astfel am folosit limbajul PDDL, care generează toate stările unei probleme, ce are un domeniu unic.

Componentele unei sarcini de planificare în PDDL sunt:

- Objects = lucruri din lumea reală care ne interesează
- Predicates = proprietăți ale obiectelor care ne interesează
- Initial state = starea lumii în care începem
- Goal specification = lucruri care vrem să fie adevărate
- Actions/Operators = modalități de schimbare a stării lumii

O problemă de planificare este creată prin asocierea unei descrieri de domeniu cu o descriere a problemei. Condițiile pre și post ale acțiunii sunt exprimate ca propoziții logice construite din predicate și termeni de argumente și conectivități logice.

Sarcinile de planificare specificate în PDDL sunt separate în două fișiere diferite:

- Fișier de domeniu care conține predicatele și acțiunile
- Fișier cu probleme care conține obiectele, starea inițială și specificația goal-ului

Structura fișierului de domeniu este următoarea:

```
(define (domain <domain name> )  
(:predicates <predicate>)  
(:action <prima actiune>)  
.....  
(:action <ultima actiune> )
```

Structura fișierului de probleme este următoarea:

```
(define (problem <problem name>)  
(:domain <domain name>)  
(:objects <obiecte>)  
(:init <starea initiala>)  
(:goal <starea finala> )
```

2 Cerința problemei

Am ales să implementăm două tipuri de domenii, unul cu aplicabilitate în lumea reală, și unul fără. Fiecare domeniu are ca scop realizarea unei rețete. Astfel, prima problemă are ca și scop final realizarea unei prăjituri cu măr, iar cea de a doua are ca scop realizarea unei pizza.

Realizarea unei rețete necesită urmarea unor pași cum ar fi: cumpărarea ingredientelor necesare, realizarea compoziției.

Problemele definite de domeniul cu aplicabilitate în lumea reală prezintă o stare inițială parțial definită, acțiunile având efecte nedeterminate, și anume nu știm dacă un anumit ingredient e stricat sau nu, urmând ca programul să determine care ingredient e stricat și care nu.

Problemele fără aplicabilitate în lumea reală nu întâlnesc aceste situații ale mediului parțial observabil, aici fiind foarte clar care ingrediente sunt stricate și care nu.



Figure 1: Goal-ul pentru problema 1.



Figure 2: Goal-ul pentru problema 2.

3 Implementare

Pentru realizarea acestor cerințe este nevoie de a defini domeniul și diferite probleme. Domeniul descriind cum arata mediul în care se desfășoară acțiunile, obiectele (agenții) care fac acțiunile și acțiunile care se petrec în acest mediu. Având un domeniu stabilit, se pot genera diferite probleme pe baza lui. În probleme se folosesc obiectele și acțiunile declarate în domeniu, după care se urmărește crearea unei condiții inițiale din care să rezulte o anumită stare finală definită de noi.

Ambele probleme create de noi au aceeași idee de implementare, diferență fiind faptul că una are aplicabilitate în mediul real, iar alta nu.

3.1 Problema fără aplicabilitate în lumea reală

3.1.1 Reprezentare domeniu pentru rețetă

Predicatele domeniului sunt:

- nu are i1 ?i1: nu are ingredientul 1
- nu are i2 ?i2: nu are ingredientul 2
- nu are i3 ?i3: nu are ingredientul 3
- nu are i4 ?i4: nu are ingredientul 4
- nu preparat final ?fin: preparatul final nu e gata
- neincalzit ?c: cuptorul nu este încălzit
- nu are amestec ? a: amestecul pentru rețetă nu e gata
- nu in cuptor ?c ?a: amestecul nu este băgat în cuptor
- nu gata ?fin: rețeta nu e gata

Funcțiile domeniului sunt:

- calitate-i1 ?i1: returnează calitatea ingredientului 1
- calitate-i2 ?i2: returnează calitatea ingredientului 2
- calitate-i3 ?i3: returnează calitatea ingredientului 3
- calitate-i4 ?i4: returnează calitatea ingredientului 4
- total-cost: calculează costul total

Acțiunile domeniului sunt:

- cumparare ingredient1: cumpără primul ingredient
- cumparare ingredient2: cumpără al doilea ingredient
- cumparare ingredient3: cumpără al treilea ingredient
- cumparare ingredient4: cumpără al patrulea ingredient
- start reteta: aici începe să se prepare rețeta
- amestec ingrediente: amestecă ingredientele pentru a obține aluatul
- incalzire cuptor: încălzește cuptorul pentru a adăuga aluatul
- gata final: preparatul este gata și pregătit pentru servire

Code:

```
1 (define (domain reteta)
2   (:requirements :strips :action-costs)
3   (:types ingredient1 ingredient2 ingredient3 ingredient4 final cuptor amestec)
4   (:predicates
5     (nu_are_i1 ?i1 - ingredient1)
```

```

6      (nu_are_i2 ?i2 - ingredient2)
7      (nu_are_i3 ?i3 - ingredient3)
8      (nu_are_i4 ?i4 - ingredient4)
9      (nu_preparat_final ?fin - final)
10     (neincalzit ?c - cuptor)
11     (nu_are_amestec ?a - amestec)
12     (nu_in_cuptor ?c - cuptor ?a - amestec)
13     (nu_gata ?fin - final)
14 )
15
16 (:functions
17   (calitate-i1 ?i1 - ingredient1) - number
18   (calitate-i2 ?i2 - ingredient2) - number
19   (calitate-i3 ?i3 - ingredient3) - number
20   (calitate-i4 ?i4 - ingredient4) - number
21   (total-cost) - number
22 )
23
24
25 (:action cumparare-ingredient1
26   :parameters (?i1 - ingredient1)
27   :precondition (and (nu_are_i1 ?i1)) ;nu are i1
28   :effect (and (not(nu_are_i1 ?i1)) ;are i1
29             (increase (total-cost) (calitate-i1 ?i1))
30             )
31 )
32 (:action cumparare-ingredient2
33   :parameters (?i2 - ingredient2)
34   :precondition (and (nu_are_i2 ?i2)) ;nu are i2
35   :effect (and (not(nu_are_i2 ?i2)) ;are i2
36             (increase (total-cost) (calitate-i2 ?i2))
37             )
38 )
39 (:action cumparare-ingredient3
40   :parameters (?i3 - ingredient3)
41   :precondition (and (nu_are_i3 ?i3)) ;nu are i3
42   :effect (and (not(nu_are_i3 ?i3)) ;are i3
43             (increase (total-cost) (calitate-i3 ?i3))
44             )
45 )
46 (:action cumparare-ingredient4
47   :parameters (?i4 - ingredient4)
48   :precondition (and (nu_are_i4 ?i4)) ;nu are i4
49   :effect (and (not(nu_are_i4 ?i4)) ;are i4
50             (increase (total-cost) (calitate-i4 ?i4))
51             )
52 )
53
54
55 (:action start_reteta
56   :parameters (?i1 - ingredient1 ?i2 - ingredient2 ?i4 - ingredient4
57               ?i3 - ingredient3 ?fin - final)
58   :precondition (and (not (nu_are_i1 ?i1)) ;are i1
59                     (not (nu_are_i2 ?i2)) ;are i2
60                     (not (nu_are_i4 ?i4)) ;are i4
61                     (not (nu_are_i3 ?i3)) ;are i3

```

```

62         (nu_preparat_final ?fin)) ;nu are produsul final
63
64     :effect(and (not (nu_preparat_final ?fin)) ;incepe prepararea produsului final
65                )
66 )
67
68 (:action amestec_ingrediente
69   :parameters (?i1 - ingredient1 ?i2 - ingredient2 ?i4 - ingredient4
70                ?i3 - ingredient3 ?a - amestec ?fin - final)
71   :precondition (and (not (nu_are_i1 ?i1)) ;are i1
72                      (not (nu_are_i2 ?i2)) ;are i2
73                      (not (nu_are_i3 ?i3)) ;are i3
74                      (not (nu_are_i4 ?i4)) ;are i4
75                      (not (nu_preparat_final ?fin)) ;produsul final se prepara
76                      (nu_are_amestec ?a) ;nu are amestec
77                )
78   :effect(and (nu_are_i1 ?i1) ;s-a consumat i1
79              (nu_are_i2 ?i2) ;s-a consumat i2
80              (nu_are_i3 ?i3) ;s-a consumat i3
81              (nu_are_i4 ?i4) ;s-a consumat i4
82              (not (nu_are_amestec ?a)) ;are amestec
83            )
84 )
85
86 (:action incalzire_cuptor
87   :parameters (?c - cuptor ?a - amestec ?fin - final)
88   :precondition (and (neincalzit ?c) ;cuptor neincalzit
89                      (not (nu_are_amestec ?a)) ;are amestec
90                      (not (nu_preparat_final ?fin)) ;produsul final se prepara
91                      (nu_in_cuptor ?c ?a) ;amestecul nu e in cuptor
92                )
93   :effect
94       (and
95         (not (neincalzit ?c)) ;cuptor incalzit
96         (nu_are_amestec ?a) ;amestecul se consuma
97         (nu_preparat_final ?fin) ;produsul nu se mai prepara
98         (not (nu_in_cuptor ?c ?a)) ;amestecul e in cuptor
99       )
100
101
102 )
103
104 (:action gata_final
105   :parameters (?c - cuptor ?a - amestec ?p - final)
106   :precondition
107       (and
108         (not (nu_in_cuptor ?c ?a)) ;amestecul e in cuptor
109         (not (neincalzit ?c)) ;cuptor incalzit
110       )
111   :effect
112       (and
113         (neincalzit ?c) ;cuptorul e stins
114         (nu_in_cuptor ?c ?a) ;amestecul nu mai e in cuptor
115         (not (nu_gata ?p)) ;gata produsul final
116       ))

```

Explicații:

- La linia 3, pentru generalitate, în loc să specificăm un anumit tip de ingredient (de exemplu ouă sau brânză), am declarat tipul ingredientX, X fiind de la 1 la 4. Asta înseamnă că, pentru acest domeniu, avem nevoie obligatoriu de un număr de 4 ingrediente pentru a realiza rețeta. De asemenea, final și amestec au fost puse tot pentru generalitate (în problemă, tipul final se referă la pizza, prăjitură etc., iar amestec la aluat, pastă etc.).
- La liniile 16-22 sunt funcțiile domeniului. Avem o funcție de calitate pentru fiecare ingredient și funcția total-cost pe care îl folosim pentru selectarea produselor ce vor fi cumpărate.
- La liniile 25-52 are loc cumpărarea ingredientelor. Aici crește totodată și costul total în funcție de calitatea fiecărui produs. Ingredientele vor fi alese în funcție de euristici.
- La liniile 55-66 este acțiunea start rețeta. Aceasta începe prepararea produsului final doar dacă are toate ingredientele la îndemână.
- La liniile 68-84 este acțiunea amestec ingrediente. Aici se consumă ingredientele pentru a obține un amestec.
- La liniile 86-102 are loc acțiunea încălzire cuptor. Aici, amestecul se pune în cuptor și se încălzește
- La liniile 104-117 are loc acțiunea gata final, unde se scoate amestecul din cuptor, și se generează produsul final.

Comanda pentru rularea codului din terminal: prover9 -f "aut.in"

3.1.2 Reprezentare problema 1

Prima problemă reprezintă realizarea unei rețete pentru o prăjitură cu măr. Obiectele problemei sunt:

- ou1, ou2, ou3, ou4: reprezintă primul ingredient
- mar1, mar2, mar3: reprezintă al doilea ingredient
- lapte1, lapte2: reprezintă al treilea ingredient
- faina: reprezintă al patrulea ingredient
- prajitura: reprezintă produsul care trebuie să rezulte în urma realizării rețetei
- cuptor: reprezintă cuptorul unde va fi adăugat aluatul pentru prăjitură
- aluat: reprezintă aluatul format în urma amestecării tuturor ingredientelor

Code:

```
1 (define (problem reteta-prajitura-mere)
2 (:domain reteta)
3 (:objects
4   ou1 ou2 ou3 ou4 - ingredient1
5   mar1 mar2 mar3 - ingredient2
6   lapte1 lapte2 - ingredient3
7   faina - ingredient4
8   prajitura - final
9   cuptor - cuptor
10  aluat - amestec
11 )
12
13 (:init
14
15   (nu_are_i1 ou1)
16   (nu_are_i1 ou2)
17   (nu_are_i1 ou3)
```

```

18      (nu_are_i1 ou4)
19      (nu_are_i2 mar1)
20      (nu_are_i2 mar2)
21      (nu_are_i2 mar3)
22      (nu_are_i3 lapte1)
23      (nu_are_i3 lapte2)
24      (nu_are_i4 faina)
25      (nu_preparat_final prajitura)
26      (neincalzit cuptor)
27      (nu_are_amestec aluat)
28      (nu_in_cuptor cuptor aluat)
29      (nu_gata prajitura)
30
31
32      (= (calitate-i1 ou1) 5) ;cel mai rau
33      (= (calitate-i1 ou2) 4)
34      (= (calitate-i1 ou3) 2) ;cel mai bun
35      (= (calitate-i1 ou4) 4)
36
37      (= (calitate-i2 mar1) 10) ;cel mai rau
38      (= (calitate-i2 mar2) 6)
39      (= (calitate-i2 mar3) 4) ;cel mai bun
40
41
42      (= (calitate-i3 lapte1) 3)
43      (= (calitate-i3 lapte2) 4) ;cel mai rau
44
45      (= (calitate-i4 faina) 1)
46
47 )
48
49 (:goal
50   (and
51     (not (nu_gata prajitura))
52   )
53 )
54 (:metric minimize (total-cost))
55 )

```

Explicații:

- Liniile 13-47 reprezintă starea inițială a problemei. În această stare, o persoană nu are nici un ingredient la dispoziție, cuptorul este neîncalzit, iar prajitura nu e gata
- Liniile 32-45 prezintă calitatea ingredientelor cumpărate. Un ingredient e de calitate dacă numărul asociat lui este cât mai mic. De exemplu, mărul 3 din magazin (calitate = 4) este mai bun decât mărul 1 (calitate = 10). Acest lucru este important deoarece, precum se vede la linia 66, pe lângă scopul de a se face prăjitura, aceasta trebuie să fie cea mai bună, adică să fie formată din ingredientele cele mai bune (suma ingredientelor să fie minimă).
- Linia 51 reprezintă scopul problemei, adică să fie gata prajitura (evidențiat prin predicatul negat (not (nu gata prajitura)).

3.1.3 Reprezentare problema 2

A doua problemă reprezintă realizarea unei rețete pentru o pizza. Obiectele problemei sunt:

- branza1, branza2: reprezintă primul ingredient
- masline1, masline2, masline3, masline4: reprezintă al doilea ingredient
- salam: reprezintă al treilea ingredient
- ciuperci: reprezintă al patrulea ingredient
- pizza: reprezintă produsul care trebuie să rezulte în urma realizării rețetei
- cuptor: reprezintă cuptorul unde va fi adăugat aluatul pentru pizza
- aluat: reprezintă aluatul format în urma amestecării tuturor ingredientelor

Code:

```
1 (define (problem reteta-prajitura-pizza)
2 (:domain reteta)
3 (:objects
4   branza1 branza2 - ingredient1
5   masline1 masline2 masline3 masline4 - ingredient2
6   salam - ingredient3
7   ciuperci - ingredient4
8   pizza - final
9   cuptor - cuptor
10  aluat - amestec
11 )
12
13 (:init
14
15   (nu_are_i1 branza1)
16   (nu_are_i1 branza2)
17   (nu_are_i2 masline1)
18   (nu_are_i2 masline2)
19   (nu_are_i2 masline3)
20   (nu_are_i2 masline4)
21   (nu_are_i3 salam)
22   (nu_are_i4 ciuperci)
23   (nu_preparat_final pizza)
24   (neincalzit cuptor)
25   (nu_are_amestec aluat)
26   (nu_in_cuptor cuptor aluat)
27   (nu_gata pizza)
28
29
30   (= (calitate-i1 branza1) 2) ;cel mai bun
31   (= (calitate-i1 branza2) 4) ;cel mai rau
32
33   (= (calitate-i2 masline1) 1) ;cel mai bun
34   (= (calitate-i2 masline2) 4)
35   (= (calitate-i2 masline3) 1) ;cel mai bun
36   (= (calitate-i2 masline4) 8) ;cel mai rau
37
38
39   (= (calitate-i3 salam) 1)
40   (= (calitate-i4 ciuperci) 1)
41
42 )
```

```
43
44 (:goal
45   (and
46     (not (nu_gata pizza))
47   )
48 )
49 (:metric minimize (total-cost))
50 )
```

Explicații:

- Această problema e asemanatoare cu cea de la problema reteta heuristics, numai că în acest caz am ales să urmărim rețeta pentru o pizza, deci avem alte ingrediente și alte valori de calitate.

3.2 Problema cu aplicabilitate în lumea reala

3.2.1 Reprezentare domeniu pentru rețetă

Predicatele domeniului sunt:

- nu are i1 ?i1: nu are ingredientul 1
- nu are i2 ?i2: nu are ingredientul 2
- nu are i3 ?i3: nu are ingredientul 3
- nu are i4 ?i4: nu are ingredientul 4
- stricat i1 ?i1: ingredientul 1 e stricat
- stricat i2 ?i2: ingredientul 2 e stricat
- stricat i3 ?i3: ingredientul 3 e stricat
- nu preparat final ?fin: preparatul final nu e gata
- neincalzit ?c: cuptorul nu este încălzit
- nu are amestec ? a: amestecul pentru rețetă nu e gata
- nu in cuptor ?c ?a: amestecul nu este băgat în cuptor
- nu gata ?fin: rețeta nu e gata

Acțiunile domeniului sunt:

- sense stricat i1: verifică dacă ingredientul 1 e stricat
- sense stricat i2: verifică dacă ingredientul 2 e stricat
- sense stricat i3: verifică dacă ingredientul 3 e stricat
- cumparare ingredient1: cumpără primul ingredient
- cumparare ingredient2: cumpără al doilea ingredient
- cumparare ingredient3: cumpără al treilea ingredient
- cumparare ingredient4: cumpără al patrulea ingredient
- arunc i1 stricat: dacă ingredientul 1 e stricat, se aruncă
- arunc i2 stricat: dacă ingredientul 2 e stricat, se aruncă
- arunc i3 stricat: dacă ingredientul 3 e stricat, se aruncă
- start retea: aici începe să se prepare produsul
- amestec ingrediente: amestecă ingredientele pentru a obține aluatul
- incalzire cuptor: încălzește cuptorul pentru a adăuga aluatul
- gata final: preparatul este gata și pregătit pentru servire

Code:

```
1 (define (domain reteta)
2 (:types ingredient1 ingredient2 ingredient3 ingredient4 final cuptor amestec)
3 (:predicates
4   (nu_are_i1 ?i1 - ingredient1)
5   (nu_are_i2 ?i2 - ingredient2)
6   (nu_are_i3 ?i3 - ingredient3)
7   (nu_are_i4 ?i4 - ingredient4)
8
9   (stricat_i1 ?i1 - ingredient1)
10  (stricat_i2 ?i2 - ingredient2)
11  (stricat_i3 ?i3 - ingredient3)
12  (nu_preparat_final ?fin - final)
13  (neincalzit ?c - cuptor))
```

```

14      (nu_are_amestec ?a - amestec)
15      (nu_in_cuptor ?c - cuptor ?a - amestec)
16      (nu_gata ?fin - final)
17  )
18
19  (:action sense_stricat_i1
20   :parameters (?i1 - ingredient1)
21   :precondition (and (not (nu_are_i1 ?i1))) ;are i1 la el
22   :observe (stricat_i1 ?i1)
23  )
24
25  (:action sense_stricat_i2
26   :parameters (?i2 - ingredient2)
27   :precondition (and (not (nu_are_i2 ?i2))) ;are i2 la el
28   :observe (stricat_i2 ?i2)
29  )
30
31
32
33  (:action sense_stricat_i3
34   :parameters (?i3 - ingredient3)
35   :precondition (and (not (nu_are_i3 ?i3))) ;are i3 la el
36   :observe (stricat_i3 ?i3)
37  )
38
39
40
41  (:action cumparare-ingredient1
42   :parameters (?i1 - ingredient1)
43   :precondition (and (nu_are_i1 ?i1)) ;nu are i1
44   :effect (and (not(nu_are_i1 ?i1))) ;are i1
45  )
46  (:action cumparare-ingredient2
47   :parameters (?i2 - ingredient2)
48   :precondition (and (nu_are_i2 ?i2)) ;nu are i2
49   :effect (and (not(nu_are_i2 ?i2))) ;are i2
50  )
51  (:action cumparare-ingredient3
52   :parameters (?i3 - ingredient3)
53   :precondition (and (nu_are_i3 ?i3)) ;nu are i3
54   :effect (and (not(nu_are_i3 ?i3))) ;are i3
55  )
56  (:action cumparare-ingredient4
57   :parameters (?i4 - ingredient4)
58   :precondition (and (nu_are_i4 ?i4)) ;nu are i4
59   :effect (and (not(nu_are_i4 ?i4))) ;are i4
60  )
61
62
63  (:action arunc_i1_stricat
64   :parameters (?i1 - ingredient1)
65   :precondition (and (not (nu_are_i1 ?i1)) ;are i1
66                   (stricat_i1 ?i1) ) ;e stricat
67   :effect(nu_are_i1 ?i1) ;arunca i1
68  )
69

```

```

70 (:action arunc_i2_stricat
71     :parameters (?i2 - ingredient2)
72     :precondition (and (not (nu_are_i2 ?i2)) ;are i2
73         (stricat_i2 ?i2) ) ;e stricat
74     :effect(nu_are_i2 ?i2) ;arunca i2
75 )
76
77 (:action arunc_i3_stricat
78     :parameters (?i3 - ingredient3)
79     :precondition (and (not (nu_are_i3 ?i3)) ;are i3
80         (stricat_i3 ?i3) ) ;e stricat
81     :effect(nu_are_i3 ?i3) ;arunca i3
82 )
83
84
85
86 (:action start_reteta
87     :parameters (?i1 - ingredient1 ?i2 - ingredient2 ?i4 - ingredient4
88         ?i3 - ingredient3 ?fin - final)
89     :precondition (and (not (nu_are_i1 ?i1)) ;are i1
90         (not (nu_are_i2 ?i2)) ;are i2
91         (not (nu_are_i4 ?i4)) ;are i4
92         (not (nu_are_i3 ?i3)) ;are i3
93         (nu_preparat_final ?fin) ;nu are produsul final
94         (not (stricat_i1 ?i1)) ;nu e i1 stricat
95         (not (stricat_i2 ?i2)) ;nu e i2 stricat
96         (not (stricat_i3 ?i3)) ;nu e i3 stricat
97     :effect(and (not (nu_preparat_final ?fin)) ;incepe prepararea produsului final
98         )
99 )
100
101 (:action amestec_ingrediente
102     :parameters (?i1 - ingredient1 ?i2 - ingredient2 ?i4 - ingredient4
103         ?i3 - ingredient3 ?a - amestec ?fin - final)
104     :precondition (and (not (nu_are_i1 ?i1)) ;are i1
105         (not (nu_are_i2 ?i2)) ;are i2
106         (not (nu_are_i3 ?i3)) ;are i3
107         (not (nu_are_i4 ?i4)) ;are i4
108         (not (nu_preparat_final ?fin)) ;produsul final se prepara
109         (nu_are_amestec ?a) ;nu are amestec
110     )
111     :effect(and (nu_are_i1 ?i1) ;s-a consumat i1
112         (nu_are_i2 ?i2) ;s-a consumat i2
113         (nu_are_i3 ?i3) ;s-a consumat i3
114         (nu_are_i4 ?i4) ;s-a consumat i4
115         (not (nu_are_amestec ?a)) ;are amestec
116     )
117 )
118
119 (:action incalzire_cuptor
120     :parameters (?c - cuptor ?a - amestec ?fin - final)
121     :precondition (and (neincalzit ?c) ;cuptor neincalzit
122         (not (nu_are_amestec ?a)) ;are amestec
123         (not (nu_preparat_final ?fin)) ;produsul final se prepara
124         (nu_in_cuptor ?c ?a) ;amestecul nu e in cuptor
125     )

```

```

126     :effect
127         (and
128             (not (neincalzit ?c)) ;cuptor incalzit
129             (nu_are_amestec ?a) ;amestecul se consuma
130             (nu_preparat_final ?fin) ;produsul nu se mai prepara
131             (not (nu_in_cuptor ?c ?a)) ;amestecul e in cuptor
132         )
133
134
135 )
136
137 (:action gata_final
138     :parameters (?c - cuptor ?a - amestec ?p - final)
139     :precondition
140         (and
141             (not (nu_in_cuptor ?c ?a)) ;amestecul e in cuptor
142             (not (neincalzit ?c)) ;cuptor incalzit
143         )
144     :effect
145         (and
146             (neincalzit ?c) ;cuptorul e stins
147             (nu_in_cuptor ?c ?a) ;amestecul nu mai e in cuptor
148             (not (nu_gata ?p)) ;gata produsul final
149         )
150 )
151
152 )

```

Explicații:

- La liniile 9-11 se remarcă 3 predicate care anunță dacă un ingredient e stricat. Asta înseamnă că pentru o problemă, numărul maxim de ingrediente care pot să fie stricate este 3 (una pentru fiecare tip de ingredient, în afară de unul singur).
- La liniile 19-37 se observă elementul predicatul necunoscut stricat ingredient pentru a testa dacă are trebuie aruncat sau nu. îl folosim pentru selectarea produselor ce vor fi cumpărate. De asemenea, condiția pentru a începe observarea este să fi cumpărat ingredientul.
- La liniile 63-82 are loc acțiunea de aruncare a ingredientelor dacă sunt stricate. După efectuarea acțiunii, se pierde ingredientul și trebuie cumpărat altul.
- La liniile 86-99, acțiunea start reteta, pe lângă faptul că trebuie să aibă câte una din fiecare tip de ingredient la dispoziție, acestea nu trebuie să fie stricate.
- La liniile 101-117 este acțiunea amestec ingrediente. Aici se consumă ingredientele pentru a obține un amestec.
- La liniile 119-135 are loc acțiunea incalzire cuptor. Aici, amestecul se pune în cuptor și se încălzește
- La liniile 137-150 are loc acțiunea gata final, unde se scoate amestecul din cuptor, și se generează produsul final.

3.2.2 Reprezentare problema 1

Prima problema reprezintă realizarea unei rețete pentru o prăjitură cu măr. Obiectele problemei sunt:

- ou1, ou2, ou3, ou4: reprezintă primul ingredient
- mar1, mar2, mar3: reprezintă al doilea ingredient
- lapte1, lapte2: reprezintă al treilea ingredient
- faina: reprezintă al patrulea ingredient
- prajitura: reprezintă produsul care trebuie să rezulte în urma realizării rețetei
- cuptor: reprezintă cuptorul unde va fi adăugat aluatul pentru prăjitură
- aluat: reprezintă aluatul format în urma amestecării tuturor ingredientelor

```
1 (define (problem reteta-prajitura-mere)
2 (:domain reteta)
3 (:objects
4   ou1 ou2 ou3 ou4 - ingredient1
5   mar1 mar2 mar3 - ingredient2
6   lapte1 lapte2 - ingredient3
7   faina - ingredient4
8   prajitura - final
9   cuptor - cuptor
10  aluat - amestec
11 )
12
13 (:init
14
15   (nu_are_i1 ou1)
16   (nu_are_i1 ou2)
17   (nu_are_i1 ou3)
18   (nu_are_i1 ou4)
19   (nu_are_i2 mar1)
20   (nu_are_i2 mar2)
21   (nu_are_i2 mar3)
22   (nu_are_i3 lapte1)
23   (nu_are_i3 lapte2)
24   (nu_are_i4 faina)
25   (nu_preparat_final prajitura)
26   (neincalzit cuptor)
27   (nu_are_amestec aluat)
28   (nu_in_cuptor cuptor aluat)
29   (nu_gata prajitura)
30
31
32   (unknown (stricat_i1 ou1))
33   (unknown (stricat_i1 ou2))
34   (unknown (stricat_i1 ou3))
35   (unknown (stricat_i1 ou4))
36   (oneof (stricat_i1 ou1) (stricat_i1 ou2) (stricat_i1 ou3) (stricat_i1 ou4))
37   (unknown (stricat_i2 mar1))
38   (unknown (stricat_i2 mar2))
39   (unknown (stricat_i2 mar3))
40   (oneof (stricat_i2 mar1) (stricat_i2 mar2) (stricat_i2 mar3))
41   (unknown (stricat_i3 lapte1))
42   (unknown (stricat_i3 lapte2))
43   (oneof (stricat_i3 lapte1) (stricat_i3 lapte2))
44
```

```

45
46 )
47
48 (:goal
49   (and
50     (not (nu_gata prajitura))
51     ;se consuma ingredientele
52     (nu_are_i1 ou1)
53     (nu_are_i1 ou2)
54     (nu_are_i1 ou3)
55     (nu_are_i1 ou4)
56     (nu_are_i2 mar1)
57     (nu_are_i2 mar2)
58     (nu_are_i2 mar3)
59     (nu_are_i3 lapte1)
60     (nu_are_i3 lapte2)
61     (nu_are_i4 faina)
62   )
63 )
64
65 )
66 )

```

Explicații:

- La liniile 32-43 am implementat ideea că un ingredient de un anumit tip din totalul de ingrediente din acel tip e stricat (de exemplu, unul din cele 4 ouă sunt stricate, sau un măr din cele 3 e stricat).
- La liniile 52-61 din scop, am evidențiat faptul că trebuie consumate toate ingredientele. Acest lucru evită situația de a cumpăra toate produsele din prima, și apoi să nu aibă loc acțiunea de arunc ingredient stricat (pentru ca nu ar mai avea rost).

3.2.3 Reprezentare problema 2

A doua problemă reprezintă realizarea unei rețete pentru o pizza. Obiectele problemei sunt:

- branza1, branza2: reprezintă primul ingredient
- masiline1, masline2, masline3, masline4: reprezintă al doilea ingredient
- salam: reprezintă al treilea ingredient
- ciuperci: reprezintă al patrulea ingredient
- pizza: reprezintă produsul care trebuie să rezulte în urma realizării rețetei
- cuptor: reprezintă cuptorul unde va fi adăugat aluatul pentru pizza
- aluat: reprezintă aluatul format în urma amestecării tuturor ingredientelor

Code:

```

1 (define (problem reteta-pizza)
2   (:domain reteta)
3   (:objects
4     branza1 branza2 - ingredient1
5     masline1 masline2 masline3 masline4 - ingredient2
6     salam - ingredient3
7     ciuperci - ingredient4
8     pizza - final
9     cuptor - cuptor
10    aluat - amestec
11  )

```



```

12
13 (:init
14
15     (nu_are_i1 branza1)
16     (nu_are_i1 branza2)
17     (nu_are_i2 masline1)
18     (nu_are_i2 masline2)
19     (nu_are_i2 masline3)
20     (nu_are_i2 masline4)
21     (nu_are_i3 salam)
22     (nu_are_i4 ciuperci)
23     (nu_preparat_final pizza)
24     (neincalzit cuptor)
25     (nu_are_amestec aluat)
26     (nu_in_cuptor cuptor aluat)
27     (nu_gata pizza)
28
29
30     (unknown (stricat_i1 branza1))
31     (unknown (stricat_i1 branza2))
32     (oneof (stricat_i1 branza1) (stricat_i1 branza2))
33     (unknown (stricat_i2 masline1))
34     (unknown (stricat_i2 masline2))
35     (unknown (stricat_i2 masline3))
36     (unknown (stricat_i2 masline4))
37     (oneof (stricat_i2 masline1) (stricat_i2 masline2)(stricat_i2 masline3)(stricat_i2 masline4))
38
39
40 )
41
42 (:goal
43     (and
44         (not (nu_gata pizza))
45         ;se consuma ingredientele
46         (nu_are_i1 branza1)
47         (nu_are_i1 branza2)
48         (nu_are_i2 masline1)
49         (nu_are_i2 masline2)
50         (nu_are_i2 masline3)
51         (nu_are_i2 masline4)
52         (nu_are_i3 salam)
53         (nu_are_i4 ciuperci)
54     )
55 )
56
57 )
58 )

```

Explicații:

- Această problemă e asemănătoare cu cea de la problem reteta contingent, numai că în acest caz am ales urmăm rețeta pentru o pizza, deci avem alte ingrediente și 2 tipuri de ingrediente stricate.

4 Rezultate

Rezultatele au fost generate, în sistemul de operare Linux, atât cu planificatorul Fast-Downward, cât și cu planificatorul Fast-Forward, în funcție de tipul problemei.

4.1 Problema fără aplicabilitate în lumea reală

Comenzi și rezultate:

Pentru a testa domeniul și cele două tipuri de problemă, ne-am folosit de planner-ul Fast-Downward și de următoarele comenzi:

```
./fast-downward.py domain_reteta_heuristics.pddl problem1_reteta_heuristics.pddl  
./fast-downward.py domain_reteta_heuristics.pddl problem2_reteta_heuristics.pddl
```

Ieșirea generată se află în fișierele output1.sas și output2.sas. Datorită lungimii lor, nu vor fi afișate aici, dar se pot observa pașii făcuți pentru atingerea scopului.

De asemenea, ne-am folosit de capabilitățile de search ale acestui planner, prin utilizarea următoarelor euristici și comenzi:

- Căutare prin A*:

```
./fast-downward.py domain_reteta_heuristics.pddl problem1_reteta_heuristics.pddl -heuristic  
"h=ff()" -search "astar(h)"  
./fast-downward.py domain_reteta_heuristics.pddl problem2_reteta_heuristics.pddl -heuristic  
"h=ff()" -search "astar(h)"
```

Mai jos este ieșirea în urma utilizării euristicii pe problema cu prăjitura. Se poate observa că A* a ales produsele de cea mai bună calitate (cele cu numărul cel mai mic), și a obținut un cost al planului de 10, care e soluția optimă.

```
1 [t=0.0160573s, 10100 KB] Actual search time: 0.00314305s  
2 cumparare-ingredient2 mar3 (4)  
3 cumparare-ingredient3 lapte1 (3)  
4 cumparare-ingredient1 ou3 (2)  
5 cumparare-ingredient4 faina (1)  
6 start_reteta ou3 mar3 faina lapte1 prajitura (0)  
7 amestec_ingredientie ou3 mar3 faina lapte1 aluat prajitura (0)  
8 incalzire_cuptor cuptor aluat prajitura (0)  
9 gata_final cuptor aluat prajitura (0)  
10 [t=0.0163102s, 10100 KB] Plan length: 8 step(s).  
11 [t=0.0163102s, 10100 KB] Plan cost: 10  
12 [t=0.0163102s, 10100 KB] Expanded 9 state(s).  
13 [t=0.0163102s, 10100 KB] Reopened 0 state(s).  
14 [t=0.0163102s, 10100 KB] Evaluated 71 state(s).  
15 [t=0.0163102s, 10100 KB] Evaluations: 71  
16 [t=0.0163102s, 10100 KB] Generated 70 state(s).  
17 [t=0.0163102s, 10100 KB] Dead ends: 0 state(s).  
18 [t=0.0163102s, 10100 KB] Expanded until last jump: 0 state(s).  
19 [t=0.0163102s, 10100 KB] Reopened until last jump: 0 state(s).  
20 [t=0.0163102s, 10100 KB] Evaluated until last jump: 1 state(s).  
21 [t=0.0163102s, 10100 KB] Generated until last jump: 0 state(s).  
22 [t=0.0163102s, 10100 KB] Number of registered states: 71  
23 [t=0.0163102s, 10100 KB] Int hash set load factor: 71/128 = 0.554688  
24 [t=0.0163102s, 10100 KB] Int hash set resizes: 7  
25 [t=0.0163102s, 10100 KB] Search time: 0.00416863s
```

```

26 [t=0.0163102s, 10100 KB] Total time: 0.0163102s
27 Solution found.
28 Peak memory: 10100 KB

```

La utilizarea euristicii pe problema cu pizza, se poate de asemenea observa că A* a ales produsele de cea mai bună calitate (cele cu numărul cel mai mic), și a obținut un cost al planului de 5.

```

1 [t=0.0126976s, 10100 KB] Actual search time: 0.00257259s
2 cumparare-ingredient1 branza1 (2)
3 cumparare-ingredient2 masline1 (1)
4 cumparare-ingredient3 salam (1)
5 cumparare-ingredient4 ciuperci (1)
6 start_reteta branza1 masline1 ciuperci salam pizza (0)
7 amestec_ingredientie branza1 masline1 ciuperci salam aluat pizza (0)
8 incalzire_cuptor cuptor aluat pizza (0)
9 gata_final cuptor aluat pizza (0)
10 [t=0.0128305s, 10100 KB] Plan length: 8 step(s).
11 [t=0.0128305s, 10100 KB] Plan cost: 5
12 [t=0.0128305s, 10100 KB] Expanded 9 state(s).
13 [t=0.0128305s, 10100 KB] Reopened 0 state(s).
14 [t=0.0128305s, 10100 KB] Evaluated 55 state(s).
15 [t=0.0128305s, 10100 KB] Evaluations: 55
16 [t=0.0128305s, 10100 KB] Generated 54 state(s).
17 [t=0.0128305s, 10100 KB] Dead ends: 0 state(s).
18 [t=0.0128305s, 10100 KB] Expanded until last jump: 0 state(s).
19 [t=0.0128305s, 10100 KB] Reopened until last jump: 0 state(s).
20 [t=0.0128305s, 10100 KB] Evaluated until last jump: 1 state(s).
21 [t=0.0128305s, 10100 KB] Generated until last jump: 0 state(s).
22 [t=0.0128305s, 10100 KB] Number of registered states: 55
23 [t=0.0128305s, 10100 KB] Int hash set load factor: 55/64 = 0.859375
24 [t=0.0128305s, 10100 KB] Int hash set resizes: 6
25 [t=0.0128305s, 10100 KB] Search time: 0.00372058s
26 [t=0.0128305s, 10100 KB] Total time: 0.0128305s
27 Solution found.
28 Peak memory: 10100 KB

```

- Căutare prin Greedy (Eager):

```

./fast-downward.py domain_reteta_heuristics.pddl problem1_reteta_heuristics.pddl -heuristic
"h=ff()" -search "eager(single(h))"
./fast-downward.py domain_reteta_heuristics.pddl problem2_reteta_heuristics.pddl -heuristic
"h=ff()" -search "eager(single(h))"

```

Mai jos este ieșirea în urma utilizării euristicii pe problema cu prăjitura. Se poate observa că, fiind o strategie greedy, a ales prima calitate pe care a întâlnit-o de fiecare dată și a obținut un cost al planului de 19, care nu e soluția optimă.

```

1 [t=0.0168054s, 10100 KB] Solution found!
2 [t=0.0169503s, 10100 KB] Actual search time: 0.00655881s
3 cumparare-ingredient2 mar1 (10)
4 cumparare-ingredient3 lapte1 (3)
5 cumparare-ingredient1 ou1 (5)
6 cumparare-ingredient4 faina (1)
7 start_reteta ou1 mar1 faina lapte1 prajitura (0)
8 amestec_ingredientie ou1 mar1 faina lapte1 aluat prajitura (0)
9 incalzire_cuptor cuptor aluat prajitura (0)

```

```

10 gata_final cuptor aluat prajitura (0)
11 [t=0.0170435s, 10100 KB] Plan length: 8 step(s).
12 [t=0.0170435s, 10100 KB] Plan cost: 19
13 [t=0.0170435s, 10100 KB] Expanded 71 state(s).
14 [t=0.0170435s, 10100 KB] Reopened 0 state(s).
15 [t=0.0170435s, 10100 KB] Evaluated 191 state(s).
16 [t=0.0170435s, 10100 KB] Evaluations: 191
17 [t=0.0170435s, 10100 KB] Generated 576 state(s).
18 [t=0.0170435s, 10100 KB] Dead ends: 0 state(s).
19 [t=0.0170435s, 10100 KB] Number of registered states: 191
20 [t=0.0170435s, 10100 KB] Int hash set load factor: 191/256 = 0.746094
21 [t=0.0170435s, 10100 KB] Int hash set resizes: 8
22 [t=0.0170435s, 10100 KB] Search time: 0.00768715s
23 [t=0.0170435s, 10100 KB] Total time: 0.0170435s
24 Solution found.
25 Peak memory: 10100 KB

```

La utilizarea euristicii pe problema cu pizza, se poate de asemenea observa că Greedy a ales calitatea primelor ingrediente pe care le-a întâlnit, și a obținut un cost al planului de 5. În acest caz, prin coincidență, costul primele ingrediente au fost defapt cele care au avut calitatea cea mai bună, deci a fost generat un plan pentru costul optim.

```

1 [t=0.0119057s, 10100 KB] Actual search time: 0.00332748s
2 cumparare-ingredient1 branza1 (2)
3 cumparare-ingredient2 masline1 (1)
4 cumparare-ingredient3 salam (1)
5 cumparare-ingredient4 ciuperci (1)
6 start_reteta branza1 masline1 ciuperci salam pizza (0)
7 amestec_ingrediente branza1 masline1 ciuperci salam aluat pizza (0)
8 incalzire_cuptor cuptor aluat pizza (0)
9 gata_final cuptor aluat pizza (0)
10 [t=0.0121114s, 10100 KB] Plan length: 8 step(s).
11 [t=0.0121114s, 10100 KB] Plan cost: 5
12 [t=0.0121114s, 10100 KB] Expanded 33 state(s).
13 [t=0.0121114s, 10100 KB] Reopened 0 state(s).
14 [t=0.0121114s, 10100 KB] Evaluated 90 state(s).
15 [t=0.0121114s, 10100 KB] Evaluations: 90
16 [t=0.0121114s, 10100 KB] Generated 186 state(s).
17 [t=0.0121114s, 10100 KB] Dead ends: 0 state(s).
18 [t=0.0121114s, 10100 KB] Number of registered states: 90
19 [t=0.0121114s, 10100 KB] Int hash set load factor: 90/128 = 0.703125
20 [t=0.0121114s, 10100 KB] Int hash set resizes: 7
21 [t=0.0121114s, 10100 KB] Search time: 0.00418295s
22 [t=0.0121114s, 10100 KB] Total time: 0.0121114s
23 Solution found.
24 Peak memory: 10100 KB

```

- Căutare prin Greedy (Lazy):

```

./fast-downward.py domain_reteta_heuristics.pddl problem1_reteta_heuristics.pddl -heuristic
"h=ff()" -search "lazy(single(h))"

./fast-downward.py domain_reteta_heuristics.pddl problem2_reteta_heuristics.pddl -heuristic
"h=ff()" -lazy(single(h))"

```

Mai jos este ieșirea în urma utilizării euristicii pe problema cu prăjitura. Se poate observa că, fiind o strategie greedy, a ales prima calitate pe care a întâlnit-o de fiecare dată și a obținut

un cost al planului de 19, care nu e soluția optimă. Diferența dintre această strategie și strategia Greedy (eager) este ordinea de cumpărare a ingredientelor.

```
1 [t=0.0156681s, 10100 KB] Actual search time: 0.00514013s
2 cumparare-ingredient1 ou1 (5)
3 cumparare-ingredient2 mar1 (10)
4 cumparare-ingredient3 lapte1 (3)
5 cumparare-ingredient4 faina (1)
6 start_reteta ou1 mar1 faina lapte1 prajitura (0)
7 amestec_ingredientie ou1 mar1 faina lapte1 aluat prajitura (0)
8 incalzire_cuptor cuptor aluat prajitura (0)
9 gata_final cuptor aluat prajitura (0)
10 [t=0.0157701s, 10100 KB] Plan length: 8 step(s).
11 [t=0.0157701s, 10100 KB] Plan cost: 19
12 [t=0.0157701s, 10100 KB] Expanded 94 state(s).
13 [t=0.0157701s, 10100 KB] Reopened 0 state(s).
14 [t=0.0157701s, 10100 KB] Evaluated 95 state(s).
15 [t=0.0157701s, 10100 KB] Evaluations: 95
16 [t=0.0157701s, 10100 KB] Generated 771 state(s).
17 [t=0.0157701s, 10100 KB] Dead ends: 0 state(s).
18 [t=0.0157701s, 10100 KB] Number of registered states: 95
19 [t=0.0157701s, 10100 KB] Int hash set load factor: 95/128 = 0.742188
20 [t=0.0157701s, 10100 KB] Int hash set resizes: 7
21 [t=0.0157701s, 10100 KB] Search time: 0.00544253s
22 [t=0.0157701s, 10100 KB] Total time: 0.0157701s
23 Solution found.
24 Peak memory: 10100 KB
```

La utilizarea euristicii pe problema cu pizza, se poate de asemenea observa că Greedy a ales calitatea primelor ingrediente pe care le-a întâlnit, și a obținut un cost al planului de 5. În acest caz, prin coincidență, costul primele ingrediente au fost defapt cele care au avut calitatea cea mai bună, deci a fost generat un plan pentru costul optim.

```
1 [t=0.0102062s, 10100 KB] Solution found!
2 [t=0.0103539s, 10100 KB] Actual search time: 0.00279978s
3 cumparare-ingredient2 masline1 (1)
4 cumparare-ingredient1 branza1 (2)
5 cumparare-ingredient3 salam (1)
6 cumparare-ingredient4 ciuperci (1)
7 start_reteta branza1 masline1 ciuperci salam pizza (0)
8 amestec_ingredientie branza1 masline1 ciuperci salam aluat pizza (0)
9 incalzire_cuptor cuptor aluat pizza (0)
10 gata_final cuptor aluat pizza (0)
11 [t=0.0106218s, 10100 KB] Plan length: 8 step(s).
12 [t=0.0106218s, 10100 KB] Plan cost: 5
13 [t=0.0106218s, 10100 KB] Expanded 51 state(s).
14 [t=0.0106218s, 10100 KB] Reopened 0 state(s).
15 [t=0.0106218s, 10100 KB] Evaluated 52 state(s).
16 [t=0.0106218s, 10100 KB] Evaluations: 52
17 [t=0.0106218s, 10100 KB] Generated 304 state(s).
18 [t=0.0106218s, 10100 KB] Dead ends: 0 state(s).
19 [t=0.0106218s, 10100 KB] Number of registered states: 52
20 [t=0.0106218s, 10100 KB] Int hash set load factor: 52/64 = 0.8125
21 [t=0.0106218s, 10100 KB] Int hash set resizes: 6
22 [t=0.0106218s, 10100 KB] Search time: 0.0030053s
23 [t=0.0106218s, 10100 KB] Total time: 0.0106218s
```

```
24 Solution found.
25 Peak memory: 10100 KB
```

4.2 Problema cu în lumea reală

Comenzi și rezultate:

Am folosit planner-ul Fast-Forward pentru generarea rezultatelor pentru domeniul și cele două tipuri de probleme. Comenzile sunt următoarele:

```
./Contingent-FF -o domain_reteta_contingent.pddl -f problem_reteta_contingent.pddl
./Contingent-FF -o domain_reteta_contingent.pddl -f problem2_reteta_contingent.pddl
```

Pentru problema cu prăjitura, rezultatul este următorul:

```
1  ff: found plan as follows
2  -----
3      0||0 --- CUMPARARE-INGREDIENT1 OU1 --- SON: 1||0
4  -----
5      1||0 --- CUMPARARE-INGREDIENT4 FAINA --- SON: 2||0
6  -----
7      2||0 --- SENSE_STRICAT_I1 OU1 --- TRUESON: 3||0 --- FALSESON: 3||1
8  -----
9      3||0 --- CUMPARARE-INGREDIENT1 OU4 --- SON: 4||0
10     3||1 --- CUMPARARE-INGREDIENT3 LAPTE2 --- SON: 4||1
11  -----
12     4||0 --- ARUNC_I1_STRICAT OU1 --- SON: 5||0
13     4||1 --- SENSE_STRICAT_I3 LAPTE2 --- TRUESON: 5||1 --- FALSESON: 5||2
14  -----
15     5||0 --- CUMPARARE-INGREDIENT3 LAPTE2 --- SON: 6||0
16     5||1 --- CUMPARARE-INGREDIENT3 LAPTE1 --- SON: 6||1
17     5||2 --- CUMPARARE-INGREDIENT3 LAPTE1 --- SON: 6||2
18  -----
19     6||0 --- SENSE_STRICAT_I3 LAPTE2 --- TRUESON: 7||0 --- FALSESON: 7||1
20     6||1 --- ARUNC_I3_STRICAT LAPTE2 --- SON: 7||2
21     6||2 --- ARUNC_I3_STRICAT LAPTE1 --- SON: 7||3
22  -----
23     7||0 --- CUMPARARE-INGREDIENT3 LAPTE1 --- SON: 8||0
24     7||1 --- CUMPARARE-INGREDIENT3 LAPTE1 --- SON: 8||1
25     7||2 --- CUMPARARE-INGREDIENT2 MAR3 --- SON: 8||2
26     7||3 --- CUMPARARE-INGREDIENT2 MAR3 --- SON: 8||3
27  -----
28     8||0 --- ARUNC_I3_STRICAT LAPTE2 --- SON: 9||0
29     8||1 --- ARUNC_I3_STRICAT LAPTE1 --- SON: 9||1
30     8||2 --- SENSE_STRICAT_I2 MAR3 --- TRUESON: 9||2 --- FALSESON: 9||3
31     8||3 --- SENSE_STRICAT_I2 MAR3 --- TRUESON: 9||4 --- FALSESON: 9||5
32  -----
33     9||0 --- CUMPARARE-INGREDIENT2 MAR3 --- SON: 10||0
34     9||1 --- CUMPARARE-INGREDIENT2 MAR3 --- SON: 10||1
35     9||2 --- CUMPARARE-INGREDIENT2 MAR2 --- SON: 10||2
36     9||3 --- START_RETETA OU1 MAR3 FAINA LAPTE1 PRAJITURA --- SON: 10||3
37     9||4 --- CUMPARARE-INGREDIENT2 MAR2 --- SON: 10||4
38     9||5 --- START_RETETA OU1 MAR3 FAINA LAPTE2 PRAJITURA --- SON: 10||5
39  -----
40     10||0 --- SENSE_STRICAT_I2 MAR3 --- TRUESON: 11||0 --- FALSESON: 11||1
41     10||1 --- SENSE_STRICAT_I2 MAR3 --- TRUESON: 11||2 --- FALSESON: 11||3
42     10||2 --- ARUNC_I2_STRICAT MAR3 --- SON: 11||4
```

```

43 10||3 --- AMESTEC_INGREDIENTE OU1 MAR3 FAINA LAPTE1 ALUAT PRAJITURA --- SON: 11||5
44 10||4 --- ARUNC_I2_STRICAT MAR3 --- SON: 11||6
45 10||5 --- AMESTEC_INGREDIENTE OU1 MAR3 FAINA LAPTE2 ALUAT PRAJITURA --- SON: 11||7
46 -----
47 11||0 --- CUMPARARE-INGREDIENT2 MAR2 --- SON: 12||0
48 11||1 --- START_RETETA OU4 MAR3 FAINA LAPTE1 PRAJITURA --- SON: 12||1
49 11||2 --- CUMPARARE-INGREDIENT2 MAR2 --- SON: 12||2
50 11||3 --- START_RETETA OU4 MAR3 FAINA LAPTE2 PRAJITURA --- SON: 12||3
51 11||4 --- START_RETETA OU1 MAR2 FAINA LAPTE1 PRAJITURA --- SON: 12||4
52 11||5 --- INCALZIRE_CUPTOR CUPTOR ALUAT PRAJITURA --- SON: 12||5
53 11||6 --- START_RETETA OU1 MAR2 FAINA LAPTE2 PRAJITURA --- SON: 12||6
54 11||7 --- INCALZIRE_CUPTOR CUPTOR ALUAT PRAJITURA --- SON: 12||7
55 -----
56 12||0 --- ARUNC_I2_STRICAT MAR3 --- SON: 13||0
57 12||1 --- AMESTEC_INGREDIENTE OU4 MAR3 FAINA LAPTE1 ALUAT PRAJITURA --- SON: 13||1
58 12||2 --- ARUNC_I2_STRICAT MAR3 --- SON: 13||2
59 12||3 --- AMESTEC_INGREDIENTE OU4 MAR3 FAINA LAPTE2 ALUAT PRAJITURA --- SON: 13||3
60 12||4 --- AMESTEC_INGREDIENTE OU1 MAR2 FAINA LAPTE1 ALUAT PRAJITURA --- SON: 13||4
61 12||5 --- GATA_FINAL CUPTOR ALUAT PRAJITURA --- SON: 13||-1
62 12||6 --- AMESTEC_INGREDIENTE OU1 MAR2 FAINA LAPTE2 ALUAT PRAJITURA --- SON: 13||5
63 12||7 --- GATA_FINAL CUPTOR ALUAT PRAJITURA --- SON: 13||-1
64 -----
65 13||0 --- START_RETETA OU4 MAR2 FAINA LAPTE1 PRAJITURA --- SON: 14||0
66 13||1 --- INCALZIRE_CUPTOR CUPTOR ALUAT PRAJITURA --- SON: 14||1
67 13||2 --- START_RETETA OU4 MAR2 FAINA LAPTE2 PRAJITURA --- SON: 14||2
68 13||3 --- INCALZIRE_CUPTOR CUPTOR ALUAT PRAJITURA --- SON: 14||3
69 13||4 --- INCALZIRE_CUPTOR CUPTOR ALUAT PRAJITURA --- SON: 14||4
70 13||5 --- INCALZIRE_CUPTOR CUPTOR ALUAT PRAJITURA --- SON: 14||5
71 -----
72 14||0 --- AMESTEC_INGREDIENTE OU4 MAR2 FAINA LAPTE1 ALUAT PRAJITURA --- SON: 15||0
73 14||1 --- GATA_FINAL CUPTOR ALUAT PRAJITURA --- SON: 15||-1
74 14||2 --- AMESTEC_INGREDIENTE OU4 MAR2 FAINA LAPTE2 ALUAT PRAJITURA --- SON: 15||1
75 14||3 --- GATA_FINAL CUPTOR ALUAT PRAJITURA --- SON: 15||-1
76 14||4 --- GATA_FINAL CUPTOR ALUAT PRAJITURA --- SON: 15||-1
77 14||5 --- GATA_FINAL CUPTOR ALUAT PRAJITURA --- SON: 15||-1
78 -----
79 15||0 --- INCALZIRE_CUPTOR CUPTOR ALUAT PRAJITURA --- SON: 16||0
80 15||1 --- INCALZIRE_CUPTOR CUPTOR ALUAT PRAJITURA --- SON: 16||1
81 -----
82 16||0 --- GATA_FINAL CUPTOR ALUAT PRAJITURA --- SON: 17||-1
83 16||1 --- GATA_FINAL CUPTOR ALUAT PRAJITURA --- SON: 17||-1
84 -----
85
86 tree layers: 17
87 total nr. actions: 65

```

Se poate observa arborele ce se obține din acțiunile disponibile. Datorită predicatelor unknown stricat care sunt observate de acțiunea sense_stricat, se creează ramuri, iar arborele ajunge să aibă înălțimea 17. Numărul total de acțiuni până la atingerea scopului este 65.

Pentru problema cu pizza, rezultatul este următorul:

```

1 ff: found plan as follows
2 -----
3 0||0 --- CUMPARARE-INGREDIENT3 SALAM --- SON: 1||0
4 -----
5 1||0 --- CUMPARARE-INGREDIENT4 CIUPERCI --- SON: 2||0

```

```

6 -----
7 2||0 --- CUMPARARE-INGREDIENT2 MASLINE4 --- SON: 3||0
8 -----
9 3||0 --- SENSE_STRICAT_I2 MASLINE4 --- TRUESON: 4||0 --- FALSESON: 4||1
10 -----
11 4||0 --- SENSE_STRICAT_I3 SALAM --- SON: 5||0
12 4||1 --- SENSE_STRICAT_I3 SALAM --- SON: 5||1
13 -----
14 5||0 --- ARUNC_I2_STRICAT MASLINE4 --- SON: 6||0
15 5||1 --- CUMPARARE-INGREDIENT1 BRANZA2 --- SON: 6||1
16 -----
17 6||0 --- CUMPARARE-INGREDIENT2 MASLINE3 --- SON: 7||0
18 6||1 --- SENSE_STRICAT_I1 BRANZA2 --- TRUESON: 7||1 --- FALSESON: 7||2
19 -----
20 7||0 --- CUMPARARE-INGREDIENT1 BRANZA2 --- SON: 8||0
21 7||1 --- SENSE_STRICAT_I3 SALAM --- SON: 8||1
22 7||2 --- START_RETETA BRANZA2 MASLINE4 CIUPERCI SALAM PIZZA --- SON: 8||2
23 -----
24 8||0 --- CUMPARARE-INGREDIENT1 BRANZA1 --- SON: 9||0
25 8||1 --- ARUNC_I1_STRICAT BRANZA2 --- SON: 9||1
26 8||2 --- AMESTEC_INGREDIENTE BRANZA2 MASLINE4 CIUPERCI SALAM ALUAT PIZZA --- SON: 9||2
27 -----
28 9||0 --- SENSE_STRICAT_I1 BRANZA1 --- TRUESON: 10||0 --- FALSESON: 10||1
29 9||1 --- CUMPARARE-INGREDIENT1 BRANZA1 --- SON: 10||2
30 9||2 --- INCALZIRE_CUPTOR CUPTOR ALUAT PIZZA --- SON: 10||3
31 -----
32 10||0 --- ARUNC_I1_STRICAT BRANZA1 --- SON: 11||0
33 10||1 --- START_RETETA BRANZA1 MASLINE3 CIUPERCI SALAM PIZZA --- SON: 11||1
34 10||2 --- START_RETETA BRANZA1 MASLINE4 CIUPERCI SALAM PIZZA --- SON: 11||2
35 10||3 --- GATA_FINAL CUPTOR ALUAT PIZZA --- SON: 11||-1
36 -----
37 11||0 --- START_RETETA BRANZA2 MASLINE3 CIUPERCI SALAM PIZZA --- SON: 12||0
38 11||1 --- ARUNC_I1_STRICAT BRANZA2 --- SON: 12||1
39 11||2 --- AMESTEC_INGREDIENTE BRANZA1 MASLINE4 CIUPERCI SALAM ALUAT PIZZA --- SON: 12||2
40 -----
41 12||0 --- AMESTEC_INGREDIENTE BRANZA2 MASLINE3 CIUPERCI SALAM ALUAT PIZZA --- SON: 13||0
42 12||1 --- AMESTEC_INGREDIENTE BRANZA1 MASLINE3 CIUPERCI SALAM ALUAT PIZZA --- SON: 13||1
43 12||2 --- INCALZIRE_CUPTOR CUPTOR ALUAT PIZZA --- SON: 13||2
44 -----
45 13||0 --- INCALZIRE_CUPTOR CUPTOR ALUAT PIZZA --- SON: 14||0
46 13||1 --- INCALZIRE_CUPTOR CUPTOR ALUAT PIZZA --- SON: 14||1
47 13||2 --- GATA_FINAL CUPTOR ALUAT PIZZA --- SON: 14||-1
48 -----
49 14||0 --- GATA_FINAL CUPTOR ALUAT PIZZA --- SON: 15||-1
50 14||1 --- GATA_FINAL CUPTOR ALUAT PIZZA --- SON: 15||-1
51 -----
52
53 tree layers: 15
54 total nr. actions: 34

```

Se poate observa ce se obține din acțiunile disponibile. Arborele ajunge să aibă înălțimea 15, iar numărul total de acțiuni până la atingerea scopului este 34. Pentru a vizualiza mai bine acest rezultat, am ales să îl reprezentăm grafic sub formă de arbore:

5 Concluzie

În concluzie, acest proiect ne-a ajutat să ne aprofundăm cunoștințele acumulate pe parcursul laboratoarelor. Am învățat să dezvoltăm și să planificăm probleme ce necesită o parte logică. Totodată am învățat să scriem cod în limbajul PDDL.