

## Spis treści

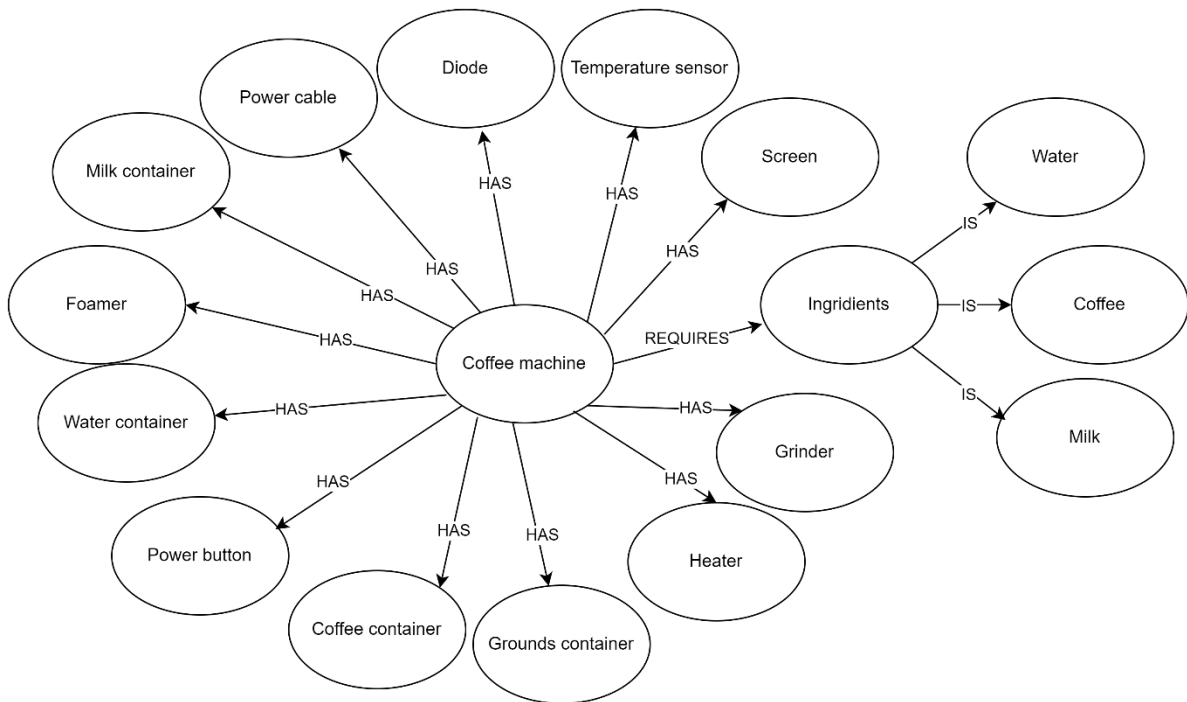
---

Sieć semantyczna .....	3
Wnioskowanie i działanie unifikacji .....	4
Reguły diagnostyczne.....	5
Baza wiedzy w Prologu.....	6
Rozwiązanie problemów za pomocą wnioskowania .....	9
Podsumowanie.....	13
Źródła .....	14

## Sieć semantyczna

W celu zapisania w Prologu prostej sieci semantycznej, należało wybrać dowolne urządzenie. W moim przypadku jest to ekspres do kawy. Na podstawie jego instrukcji, zostały opisane jego elementy składowe oraz reguły występujące w przypadku awarii urządzenia.

W celu zdefiniowania sieci semantycznej, należało określić, że ekspres do kawy jest urządzeniem, utworzyć predykaty dla składników, potrzebnych do działania urządzenia. Oprócz tego zostały utworzone predykaty, które opisują, jakie elementy składowe posiada ekspres do kawy.



Rysunek 1 Sieć semantyczna

## Wnioskowanie i działanie unifikacji

---

Opierając się na definicja unifikacji, której treść brzmi:

Unifikacja to doprowadzenie dwóch (lub w ogólnym wypadku większej liczby) formuł do identycznej postaci przez zastosowanie do nich jednego wspólnego podstawienia.

Wynika stąd, że naszym celem jest łączenie zmiennych oraz wartości w taki sposób, aby znaleźć wspólne podstawienie, które spełnia wybrane warunki.

Przykładowe zapytanie może wyglądać w następujący sposób:

ingridients(coffee) = ingridients(Ingridient) => Ingridient = coffee

Wyjaśniając powyższe działanie, ingridients(Ingridient) unifikuje się z ingridients(coffee), dlatego zmienna Ingridient zostaje związana z wartością water.

Analogiczne działanie, można przeprowadzić dla elementów, z których składa się ekspres.

Innym przykładem reguły wnioskowania, jest reguła make\_coffee. Jeżeli wybrane urządzenie posiada wymagane elementy oraz dostępne są potrzebne składniki, kawa będzie mogła zostać przygotowana.

```
make_coffee(X) :-
    device_has(X, water_container),
    device_has(X, coffee_container),
    device_has(X, milk_container),
    ingridients(coffee),
    ingridients(water),
    ingridients(milk).
```

**Rysunek 2 Przykładowa reguła wnioskowania**

## Reguły diagnostyczne

Na podstawie instrukcji obsługi ekspresu do kawy, zostały opracowane problemy, przyczyny oraz przykładowe rozwiązania w przypadku ich wystąpienia.

Problem	Przyczyna	Rozwiązanie
Ekspres do kawy nie chce się włączyć	Ekspres do kawy nie jest podłączony do prądu	Podłącz ekspres do prądu
	Wyświetlacz jest wyłączony	Włącz ekspres za pomocą przycisku zasilającego
Przygotowana kawa jest zimna	Czujnik temperatury jest zepsuty	Zgłoś usterkę do serwisu
	Grzałka w ekspresie jest zepsuta	Zgłoś usterkę do serwisu
Ekspres do kawy nie może przygotować kawy	Brak ziaren kawy w zbiorniku	Uzupełnij ziarna kawy w zbiorniku
	Brak wody w zbiorniku	Dolej wody do zbiornika
	Brak mleka w zbiorniku	Dolej mleka do zbiornika
	Pojemnik na fusy jest pełny	Opróżnij pojemnik na fusy
Wyświetlacz w ekspresie do kawy nie działa	Ekspres do kawy nie jest podłączony do prądu	Podłącz ekspres do prądu
	Wyświetlacz jest wyłączony	Włącz ekspres za pomocą przycisku zasilającego
Dioda w ekspresie do kawy miga	Brak ziaren kawy w zbiorniku	Uzupełnij ziarna kawy w zbiorniku
	Brak wody w zbiorniku	Dolej wody do zbiornika
	Brak mleka w zbiorniku	Dolej mleka do zbiornika
W przygotowanej kawie brakuje mleka	Brak mleka w zbiorniku	Dolej mleka do zbiornika
	Spieniacz do mleka jest zepsuty	Zgłoś usterkę do serwisu

## Baza wiedzy w Prologu

---

1. Zdefiniowanie faktu, że ekspres do kawy jest urządzeniem.

```
device(coffe_machine).
```

2. Zdefiniowanie składników potrzebnych do przygotowania kawy.

```
ingridients(coffee).  
ingridients(water).  
ingridients(milk).
```

3. Zdefiniowanie elementów składowych ekspresu do kawy.

```
device_element(power_cable).  
device_element(milk_container).  
device_element(water_container).  
device_element(coffee_container).  
device_element(grounds_container).  
device_element(heater).  
device_element(grinder).  
device_element(power_button).  
device_element(screen).  
device_element(foamer).  
device_element(temperature_sensor).  
device_element(diode).
```

4. Zdefiniowanie składników potrzebnych do przygotowania wybranej kawy.

```
make_coffee_milk(coffee_machine, coffee).  
make_coffee_milk(coffee_machine, water).  
make_coffee_milk(coffee_machine, milk).  
  
make_coffee_black(coffee_machine, coffee).  
make_coffee_black(coffee_machine, water).
```

5. Zdefiniowanie predykatów, które określają fakt, że ekspres do kawy posiada konkretne elementy.

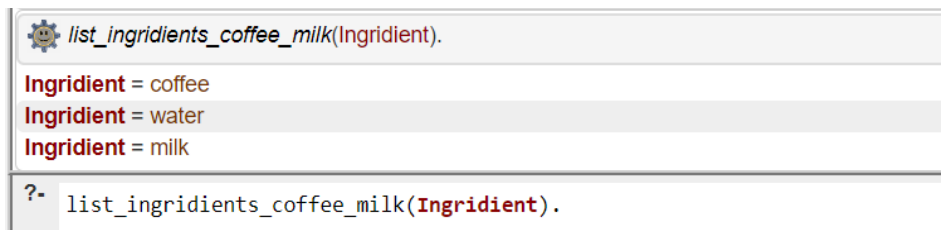
```
device_has(coffee_machine, power_cable).
device_has(coffee_machine, milk_container).
device_has(coffee_machine, water_container).
device_has(coffee_machine, coffee_container).
device_has(coffee_machine, grounds_container).
device_has(coffee_machine, heater).
device_has(coffee_machine, grinder).
device_has(coffee_machine, power_button).
device_has(coffee_machine, screen).
device_has(coffee_machine, foamer).
device_has(coffee_machine, temperature_sensor).
device_has(coffee_machine, diode).
```

W wyniku zdefiniowania powyższych faktów, możliwe było stworzenie reguł, dotyczących korzystania z ekspresu oraz opisujących jego elementy.

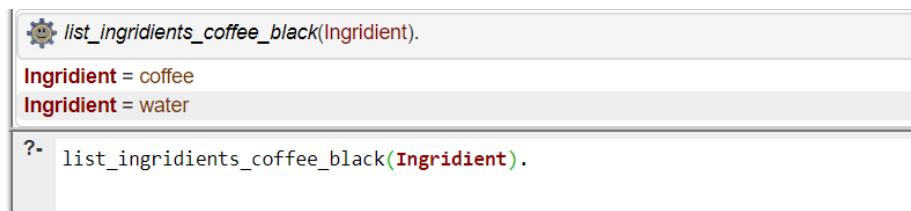
1. Zdefiniowanie reguł, określających składniki potrzebne do wykonania poszczególnej kawy.

```
list_ingredients_coffee_milk(Ingridient) :-
    make_coffee_milk(coffee_machine, Ingridient).

list_ingredients_coffee_black(Ingridient) :-
    make_coffee_black(coffee_machine, Ingridient).
```



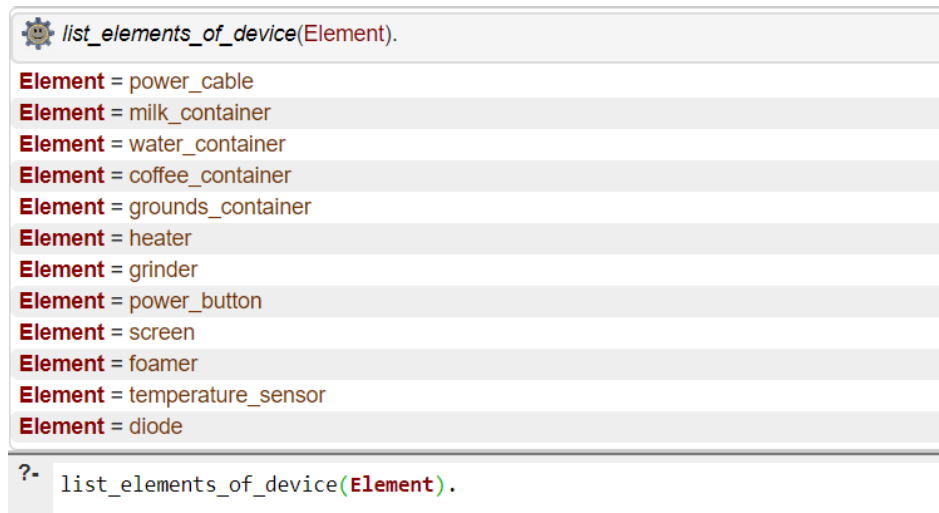
Rysunek 3 Wynik działania reguły



Rysunek 4 Wynik działania reguły

2. Zdefiniowanie reguły, określającej wszystkie elementy ekspresu do kawy.

```
list_elements_of_device(Element) :-  
    device_has(coffee_machine, Element).
```



Rysunek 5 Wynik działania reguły

## Rozwiązanie problemów za pomocą wnioskowania

Na podstawie reguł diagnostycznych, zostały opracowane fakty, dotyczące problemów, przyczyn i rozwiązań.

1. Zdefiniowanie problemów występujących podczas korzystania z ekspresu do kawy.

```
problem(coffee_machine_does_not_turn_on).
problem(cold_coffee).
problem(coffee_machine_cant_make_coffee).
problem(screen_is_not_working).
problem(diode_flashes).
problem(coffee_missing_milk).
```

2. Zdefiniowanie przyczyn, które mogą powodować wcześniej zdefiniowane problemy.

```
cause('Lack of coffee').
cause('Lack of water').
cause('Lack of milk').
cause('Full grounds container').
cause('Broken heater').
cause('Broken foamer').
cause('Broken temperature sensor').
cause('Unplugged power cable').
cause('Turned off screen').
cause('Broken coffee machine').
```

3. Zdefiniowanie rozwiązań, które mogą rozwiązywać wcześniej zdefiniowane problemy.

```
solution('Add coffee').
solution('Fill the water container').
solution('Fill the milk container').
solution('Plug in power cable').
solution('Empty grounds container').
solution('Press power button').
solution('Clean foamer').
solution('Contact service').
```

4. Zdefiniowanie przyczyn dla określonych problemów.

```
diagnosis(coffee_machine_does_not_turn_on, 'Unplugged power cable').
diagnosis(coffee_machine_does_not_turn_on, 'Turned off screen').
diagnosis(cold_coffee, 'Broken heater').
diagnosis(cold_coffee, 'Broken temperature sensor').
diagnosis(coffee_machine_cant_make_coffee, 'Lack of coffee').
diagnosis(coffee_machine_cant_make_coffee, 'Lack of water').
diagnosis(coffee_machine_cant_make_coffee, 'Lack of milk').
diagnosis(coffee_machine_cant_make_coffee, 'Full grounds container').
diagnosis(screen_is_not_working, 'Turned off screen').
diagnosis(screen_is_not_working, 'Unplugged power cable').
diagnosis(diode_flashes, 'Lack of coffee').
diagnosis(diode_flashes, 'Lack of water').
diagnosis(diode_flashes, 'Lack of milk').
diagnosis(coffee_missing_milk, 'Lack of milk').
diagnosis(coffee_missing_milk, 'Broken foamer').
```



## 5. Zdefiniowanie rozwiązań na podstawie przyczyny problemu.

```

solving_problem('Lack of coffee', 'Add coffee').
solving_problem('Lack of water', 'Fill the water container').
solving_problem('Lack of milk', 'Fill the milk container').
solving_problem('Full grounds container', 'Empty grounds container').
solving_problem('Broken heater', 'Contact service').
solving_problem('Broken foamer', 'Clean foamer').
solving_problem('Broken temperature sensor', 'Contact service').
solving_problem('Unplugged power cable', 'Plug in power cable').
solving_problem('Turned off screen', 'Press power button').
solving_problem('Broken coffee machine', 'Contact service').

```

## 6. Zdefiniowanie problemów występujących dla konkretnych elementów ekspresu do kawy.

```

possible_problems(power_button, coffee_machine_does_not_turn_on).
possible_problems(power_button, screen_is_not_working).
possible_problems(power_cable, coffee_machine_does_not_turn_on).
possible_problems(foamer, coffee_missing_milk).
possible_problems(milk_container, diode_flashes).
possible_problems(foamer, coffee_missing_milk).
possible_problems(water_container, diode_flashes).
possible_problems(coffee_container, diode_flashes).
possible_problems(heater, cold_coffee).
possible_problems(temperature_sensor, cold_coffee).
possible_problems(grounds_container, coffee_machine_cant_make_coffee).
possible_problems(screen, screen_is_not_working).

```

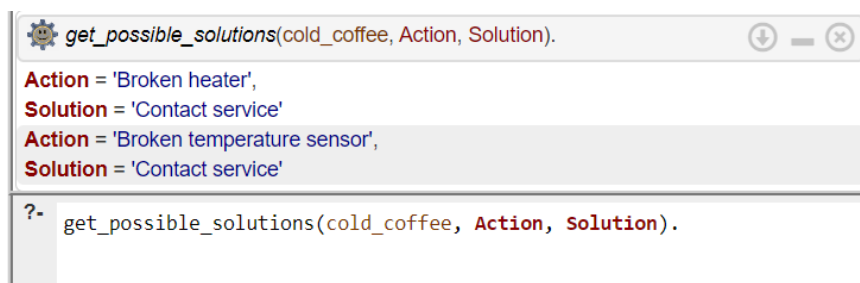
W wyniku stworzenia powyższych faktów, możliwe było stworzenie reguł, które pozwalają użytkownikowi operować na problemach występujących podczas korzystania z ekspresu do kawy.

## 1. Zdefiniowanie reguły, która na podstawie problemu zwraca nam przyczyny oraz możliwe rozwiązania.

```

get_possible_solutions(Problem, Action, Solution) :-
    diagnosis(Problem, Action), solving_problem(Action, Solution).

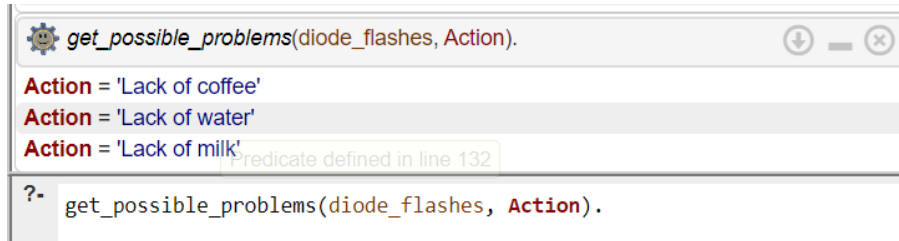
```



Rysunek 6 Wynik działania reguły

2. Zdefiniowanie reguły, która na podstawie problemu zwraca jego diagnozę.

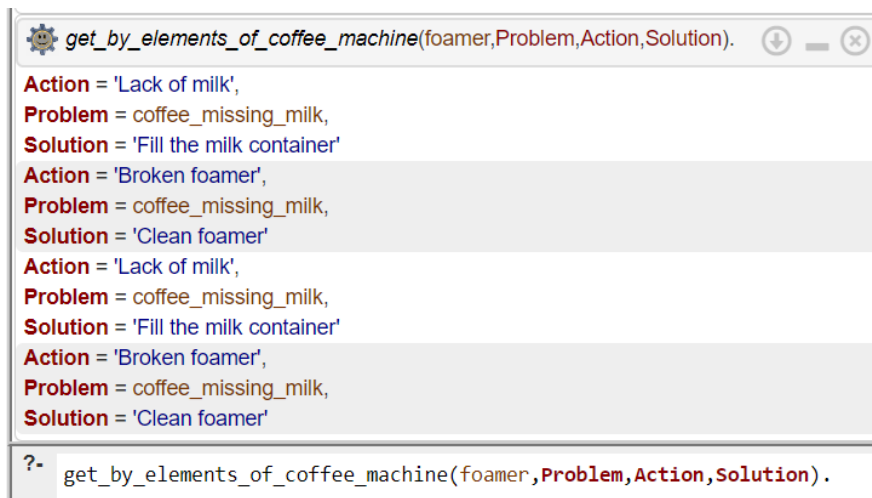
```
get_possible_problems(Problem, Action) :-  
    diagnosis(Problem, Action).
```



Rysunek 7 Wynik działania reguły

3. Zdefiniowanie reguły, która na podstawie wybranego elementu ekspresu do kawy, zwraca połączone z nimi problemy, ich przyczyny oraz możliwe rozwiązania.

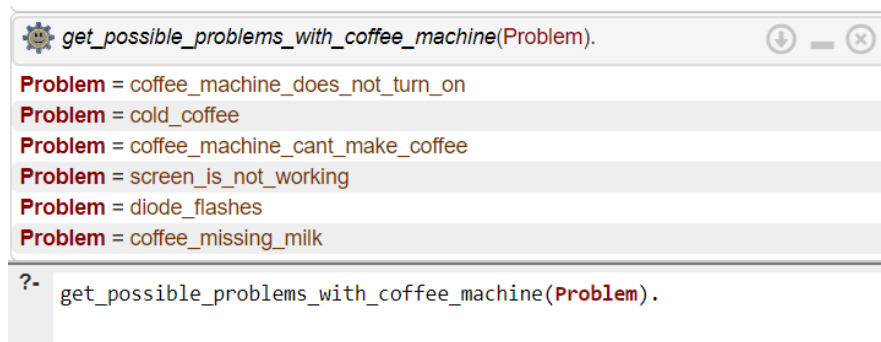
```
get_by_elements_of_coffee_machine(Part, Problem, Action, Solution) :-  
    device_has(coffee_machine, Part), possible_problems(Part, Problem),  
    diagnosis(Problem, Action), solving_problem(Action, Solution).
```



Rysunek 8 Wynik działania reguły

4. Zdefiniowanie reguły, która zwraca problemy pojawiające się podczas korzystania z ekspresu do kawy.

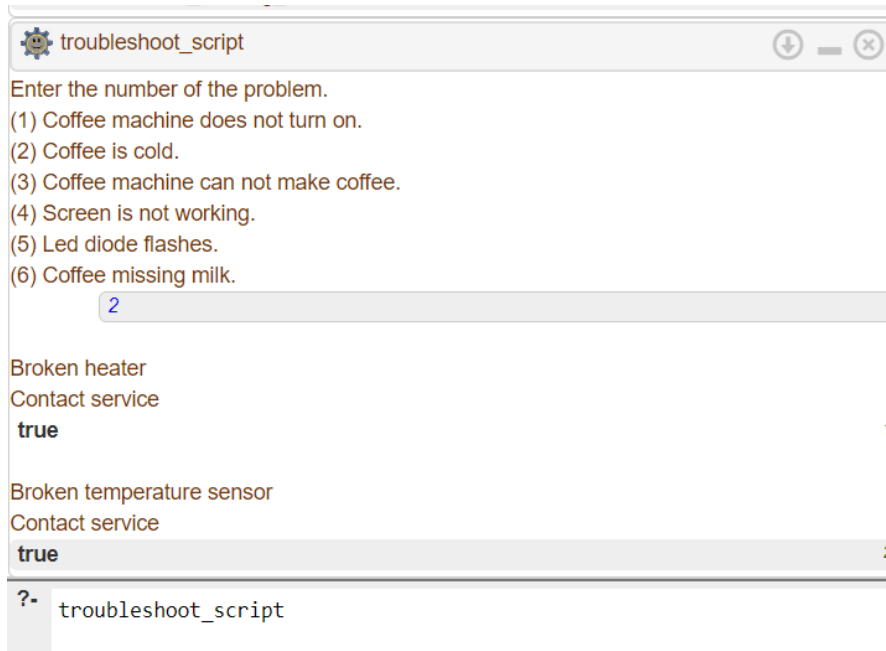
```
get_possible_problems_with_coffee_machine(Problem) :-  
    problem(Problem).
```



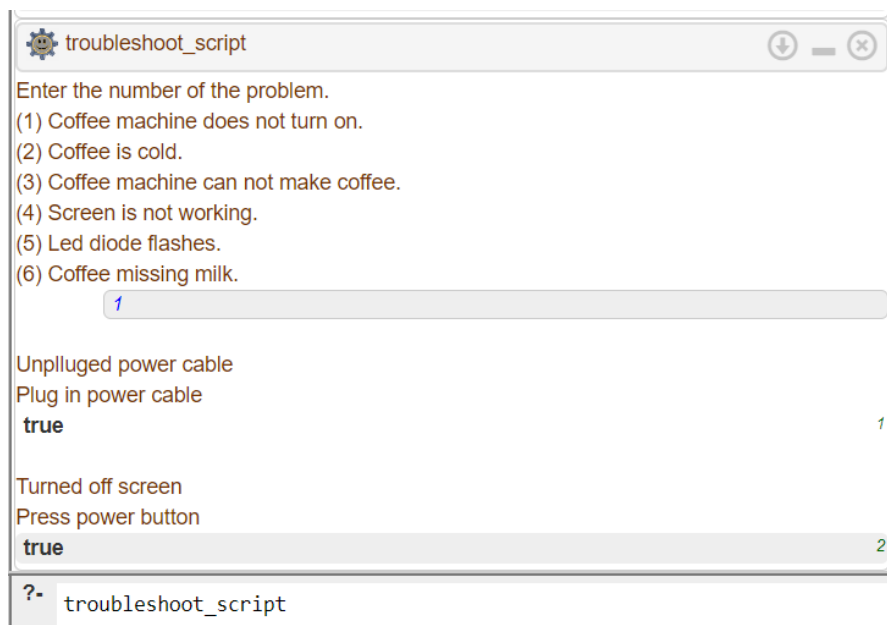
Rysunek 9 Wynik działania reguły

## Podsumowanie

W celu łatwiejszego korzystania z przygotowanych reguł zostało wprowadzone menu dla użytkownika, które po wybraniu problemu występującego na liście, zwraca możliwe przyczyny oraz rozwiązania.



Rysunek 10 Wynik działania menu



Rysunek 11 Wynik działania menu

Prolog umożliwił przygotowanie programu, który na podstawie problemów, może wykazać jego przyczyny oraz potencjalne rozwiązania, wykorzystując reguły wnioskowania.

## Źródła

---

1. <https://eportal.pwr.edu.pl/mod/folder/view.php?id=295198>
2. [https://wazniak.mimuw.edu.pl/index.php?title=Sztuczna\\_inteligencja/SI\\_Modu%C5%82\\_2\\_-\\_Od\\_logiki\\_do\\_wnioskowania#Podstawienia\\_i\\_unifikacja](https://wazniak.mimuw.edu.pl/index.php?title=Sztuczna_inteligencja/SI_Modu%C5%82_2_-_Od_logiki_do_wnioskowania#Podstawienia_i_unifikacja)
3. [https://www.youtube.com/watch?v=SykxWpFwMGs&t=1970s&ab\\_channel=DerekBanas](https://www.youtube.com/watch?v=SykxWpFwMGs&t=1970s&ab_channel=DerekBanas)
4. <https://www.delonghi.com/pl-pl/manuals/ecam610-55-sb-primadonna-soul-automatic-coffee-maker/p/ECAM610.55.SB>