

19.06.2015 r.

Laura Świś  
Damian Rusin  
Michał Janowicz  
BSI1\_Z2

## Sprawozdanie z laboratorium sztucznej inteligencji

### Informatyka, sem. VI

#### I. Opis zadania

Celem zadania było rozwiązanie problemu SRAT (Self Referential Aptitude Test), z wykorzystaniem biblioteki CLP(FD).

Test autoreferencyjny składa się z 10 pytań. Każde pytanie ma 5 możliwych odpowiedzi (od *a* do *e*). Dla każdej z 50 możliwych odpowiedzi, jest określony konkretny warunek. Rozwiązaniem testu jest funkcja przypisująca każdemu pytaniu literę (odpowieź), która spełnia określony warunek. Każde pytanie ma znaczenie i jego odpowiedź ma wpływ na rozwiązanie testu.

Jest tylko jeden prawidłowy zestaw odpowiedzi:

1:c 2:d 3:e 4:b 5:e 6:e 7:d 8:c 9:b 10:a

#### II. Założenia realizacyjne

##### 1. Założenia dodatkowe

Zadanie zrealizowano zgodnie z punktem I. Nie przyjęto żadnych dodatkowych założeń.

##### 2. Algorytmy używane do rozwiązania zadania

Algorytm zliczający liczbę wystąpień danej odpowiedzi we wszystkich pytaniach, np. ile razy występuje odpowiedź *a*.

**Dane:** *X* – numer odpowiedzi,  $[X/T]$  – lista, w której zlicza, *NR* – liczba wystąpień (*R* to numer pytania)

**Wyjście:** liczba wystąpień danej odpowiedzi

**Metoda:**

1. Jeśli lista jest pusta to zwróć 0.
2. Jeśli głowa listy równa się *X*, to zacznij zliczanie od głowy.
3. Jeśli głowa listy jest różna od *X*, to zlicz w ogonie listy.

### 3. Języki programowania, narzędzia informatyczne i środowiska używane do implementacji systemu

Problem SRAT został rozwiązany w języku programowania logicznego Prolog oraz z użyciem interfejsu SWI-Prolog.

### III. Podział prac

Autor	Podzadanie
Laura Świś	Analiza modelu matematycznego przedstawionego w książce <i>Finite Domain Constraint Programming in Oz. A Tutorial</i> , autorstwa Christiana Schulte i Gerta Smolka. Analiza skryptu w języku programowania Oz. Implementacja całej logiki odpowiedzialnej za rozwiązanie testu.
Damian Rusin	Analiza modelu matematycznego przedstawionego w książce <i>Finite Domain Constraint Programming in Oz. A Tutorial</i> , autorstwa Christiana Schulte i Gerta Smolka. Analiza skryptu w języku programowania Oz. Implementacja algorytmu zliczającego liczbę wystąpień danej odpowiedzi.
Michał Janowicz	Analiza modelu matematycznego przedstawionego w książce <i>Finite Domain Constraint Programming in Oz. A Tutorial.</i> , autorstwa Christiana Schulte i Gerta Smolka. Analiza skryptu w języku programowania Oz.

### IV. Opis implementacji

#### 1. Struktury danych

Podstawowe struktury danych to dwie tablice jednowymiarowe, które przechowują dane potrzebne do obliczeń. Dla każdej tablicy określona jest dziedzina, która oznacza, jakie minimalne i maksymalne wartości mogą przyjmować ich zmienne.

Tablica *Rozwiązanie* przechowuje odpowiedzi na dane pytania. Pytanie reprezentowane jest przez zmienną  $QN$ , gdzie  $N$  oznacza numer pytania i  $QN \in \langle 1, 5 \rangle$ .

$Rozwiazanie = [Q1, Q2, Q3, Q4, Q5, Q6, Q7, Q8, Q9, Q10],$   
*Rozwiązanie ins 1..5*

Tablica *Zliczenia* przechowuje liczbą wystąpień danej odpowiedzi w tablicy *Rozwiązanie*. Liczba wystąpień danej odpowiedzi reprezentowana jest przez zmienną  $NR$ , gdzie  $R$  oznacza numer odpowiedzi i  $NR \in \langle 0, 10 \rangle$ .

$Zliczenia = [N1, N2, N3, N4, N5]$   
*Zliczenia ins 0..10*

Pytania w programie reprezentowane są za pomocą cyfr arabskich:

- 1 – A
- 2 – B
- 3 – C
- 4 – D
- 5 – E

## 2. Funkcje i predykaty zdefiniowane w programie

### Biblioteka Constraint Logic Programming over Finite Domains

W programie użyto bibliotekę CLP(FD), która jest standardem w środowisku SWI-Prolog. Dzięki tej bibliotece możliwe jest programowanie z więzami.

```
:- use_module(library(clpfd)).
```

### Funkcja zliczająca liczbę wystąpień danej odpowiedzi

Funkcja ma za zadanie zliczyć ile razy dana odpowiedź, np. odpowiedź A, jest poprawną odpowiedzią we wszystkich 10 pytaniach.

**Dane wejściowe:** *X* - numer odpowiedzi, *Rozwiązanie* – tablica wszystkich odpowiedzi

**Dany wyjściowy:** *NR* – liczba wystąpień (*R*, oznacza numer pytania)

```
zlicz(_, [], 0) :- !.
zlicz(X, [X|T], N) :-
    zlicz(X, T, N1),
    N is N1 + 1.
zlicz(X, [_|T], N) :-
    zlicz(X, T, N).
```

### Funkcja obliczająca wynik odpowiedzi

Funkcja *srat* zawiera w sobie deklarację struktur danych oraz równania, za pomocą których zdefiniowane są zależności pomiędzy pytaniami i ich odpowiedziami. Każde pytanie jest zależne i ściśle powiązane z innymi pytaniami, i żadne z nich nie może funkcjonować pojedynczo. Prawidłowe działanie programu gwarantuje jedynie poprawna definicja wszystkich zależności. W implementacji tej funkcji wykorzystano programowanie z więzami, a także funkcję *label* dostępną dzięki bibliotece CLP(FD) oraz przedstawioną powyżej funkcję *zlicz*.

Funkcja *label* przypisuje każdej zmiennej odpowiedni wartości. Funkcja wypróbowuje wszystkie możliwe rozwiązania określone w dziedzinach, dopóki wszystkim zmiennym nie zostanie przypisana właściwa wartość.

```
srat(Rozwiązanie) :-
    Rozwiązanie=[Q1,Q2,Q3,Q4,Q5,Q6,Q7,Q8,Q9,Q10],
```

Zliczenia=[N1,N2,N3,N4,N5],  
 Rozwiazanie ins 1..5,  
 Zliczenia ins 0..10,

$N1 + N2 + N3 + N4 + N5 \# = 10,$

$Q1 \# = 1 \# \Leftrightarrow Q1 \# \setminus = 2 \# / \setminus Q2 \# = 2,$   
 $Q1 \# = 2 \# \Leftrightarrow Q2 \# \setminus = 2 \# / \setminus Q3 \# = 2,$   
 $Q1 \# = 3 \# \Leftrightarrow Q1 \# \setminus = 2 \# / \setminus Q2 \# \setminus = 2 \# / \setminus Q3 \# \setminus = 2 \# / \setminus Q4 \# = 2,$   
 $Q1 \# = 4 \# \Leftrightarrow Q1 \# \setminus = 2 \# / \setminus Q2 \# \setminus = 2 \# / \setminus Q3 \# \setminus = 2 \# / \setminus Q4 \# \setminus = 2$   
 $\# / \setminus Q5 \# = 2,$   
 $Q1 \# = 5 \# \Leftrightarrow Q1 \# \setminus = 2 \# / \setminus Q2 \# \setminus = 2 \# / \setminus Q3 \# \setminus = 2 \# / \setminus Q4 \# \setminus = 2$   
 $\# / \setminus Q5 \# \setminus = 2 \# / \setminus Q6 \# = 2,$

$Q2 \# = 1 \# \Leftrightarrow Q2 \# = Q3 \# / \setminus Q3 \# \setminus = Q4 \# / \setminus Q4 \# \setminus = Q5 \# / \setminus Q5 \# \setminus =$   
 $Q6 \# / \setminus Q6 \# \setminus = Q7,$   
 $Q2 \# = 2 \# \Leftrightarrow Q2 \# \setminus = Q3 \# / \setminus Q3 \# = Q4 \# / \setminus Q4 \# \setminus = Q5 \# / \setminus Q5 \# \setminus =$   
 $Q6 \# / \setminus Q6 \# \setminus = Q7,$   
 $Q2 \# = 3 \# \Leftrightarrow Q2 \# \setminus = Q3 \# / \setminus Q3 \# \setminus = Q4 \# / \setminus Q4 \# = Q5 \# / \setminus Q5 \# \setminus =$   
 $Q6 \# / \setminus Q6 \# \setminus = Q7,$   
 $Q2 \# = 4 \# \Leftrightarrow Q2 \# \setminus = Q3 \# / \setminus Q3 \# \setminus = Q4 \# / \setminus Q4 \# \setminus = Q5 \# / \setminus Q5 \# =$   
 $Q6 \# / \setminus Q6 \# \setminus = Q7,$   
 $Q2 \# = 5 \# \Leftrightarrow Q2 \# \setminus = Q3 \# / \setminus Q3 \# \setminus = Q4 \# / \setminus Q4 \# \setminus = Q5 \# / \setminus Q5 \# \setminus =$   
 $Q6 \# / \setminus Q6 \# = Q7,$

$Q1 \# \setminus = Q2, Q7 \# \setminus = Q8, Q8 \# \setminus = Q9, Q9 \# \setminus = Q10,$

$Q3 \# = 1 \# \Leftrightarrow Q1 \# = Q3,$   
 $Q3 \# = 2 \# \Leftrightarrow Q2 \# = Q3,$   
 $Q3 \# = 3 \# \Leftrightarrow Q4 \# = Q3,$   
 $Q3 \# = 4 \# \Leftrightarrow Q7 \# = Q3,$   
 $Q3 \# = 5 \# \Leftrightarrow Q6 \# = Q3,$

$Q4 \# = 1 \# \Leftrightarrow N1 \# = 0,$   
 $Q4 \# = 2 \# \Leftrightarrow N1 \# = 1,$   
 $Q4 \# = 3 \# \Leftrightarrow N1 \# = 2,$   
 $Q4 \# = 4 \# \Leftrightarrow N1 \# = 3,$   
 $Q4 \# = 5 \# \Leftrightarrow N1 \# = 4,$

$Q5 \# = 1 \# \Leftrightarrow Q10 \# = Q5,$   
 $Q5 \# = 2 \# \Leftrightarrow Q9 \# = Q5,$   
 $Q5 \# = 3 \# \Leftrightarrow Q8 \# = Q5,$   
 $Q5 \# = 4 \# \Leftrightarrow Q7 \# = Q5,$   
 $Q5 \# = 5 \# \Leftrightarrow Q6 \# = Q5,$

$Q6 \# = 1 \# \Leftrightarrow N1 \# = N2,$   
 $Q6 \# = 2 \# \Leftrightarrow N1 \# = N3,$   
 $Q6 \# = 3 \# \Leftrightarrow N1 \# = N4,$   
 $Q6 \# = 4 \# \Leftrightarrow N1 \# = N5,$   
 $Q6 \# = 5 \# \Leftrightarrow N1 \# \setminus = N2 \# / \setminus N1 \# \setminus = N3 \# / \setminus N1 \# \setminus = N4 \# / \setminus N1 \# \setminus =$   
 $N5,$

$Q7 \# = 1 \# \Leftrightarrow Q8 - Q7 \# = 4,$   
 $Q7 \# = 2 \# \Leftrightarrow Q8 - Q7 \# = 3,$   
 $Q7 \# = 3 \# \Leftrightarrow Q8 - Q7 \# = 2 \# \setminus / Q7 - Q8 \# = 2,$

```

Q7 #= 4 #<==> Q8 - Q7 #= 1 #\ / Q7 - Q8 #= 1,
Q7 #= 5 #<==> Q8 #= 5,

Q8 #= 1 #<==> N1 + N5 #= 2,
Q8 #= 2 #<==> N1 + N5 #= 3,
Q8 #= 3 #<==> N1 + N5 #= 4,
Q8 #= 4 #<==> N1 + N5 #= 5,
Q8 #= 5 #<==> N1 + N5 #= 6,

Q9 #= 1 #<==> (N2 + N3 + N4) #= 2 #\ / (N2 + N3 + N4) #= 3 #\ /
(N2 + N3 + N4) #= 5 #\ / (N2 + N3 + N4) #= 7,
Q9 #= 2 #<==> (N2 + N3 + N4) #= 1 #\ / (N2 + N3 + N4) #= 2 #\ /
(N2 + N3 + N4) #= 6,
Q9 #= 4 #<==> (N2 + N3 + N4) #= 0 #\ / (N2 + N3 + N4) #= 1 #\ /
(N2 + N3 + N4) #= 8,
Q9 #= 3 #<==> (N2 + N3 + N4) #= 0 #\ / (N2 + N3 + N4) #= 1 #\ /
(N2 + N3 + N4) #= 4 #\ / (N2 + N3 + N4) #= 9,
Q9 #= 5 #<==> (N2 + N3 + N4) mod 5 #= 0,

label(Zliczenia),
label(Rozwiazanie),
zlicz(1,Rozwiazanie,N1),
zlicz(2,Rozwiazanie,N2),
zlicz(3,Rozwiazanie,N3),
zlicz(4,Rozwiazanie,N4),
zlicz(5,Rozwiazanie,N5).

```

### **Funkcja *maplist***

Funkcja *maplist* jest funkcją dostępną w środowisku SWI-Prolog i pozwala uzyskać nową listę, której wartości są zdefiniowane za pomocą starej listy. W naszym programie funkcję *maplist* wykorzystujemy do dodania do każdej wartości w tablicy *Rozwiazanie* wartość 64.

```
maplist(plus(64),N,M)
```

### **Funkcja *put\_char***

Funkcja *put\_char* jest funkcją dostępną w środowisku SWI-Prolog i służy do wypisywania znaków ASCII. Za pomocą funkcji *maplist* uzyskaliśmy wartości odpowiadające właściwym znakom ASCII i dzięki temu rozwiązanie może być przedstawione za pomocą liter, a nie cyfr.

```
maplist(put_char,M)
```

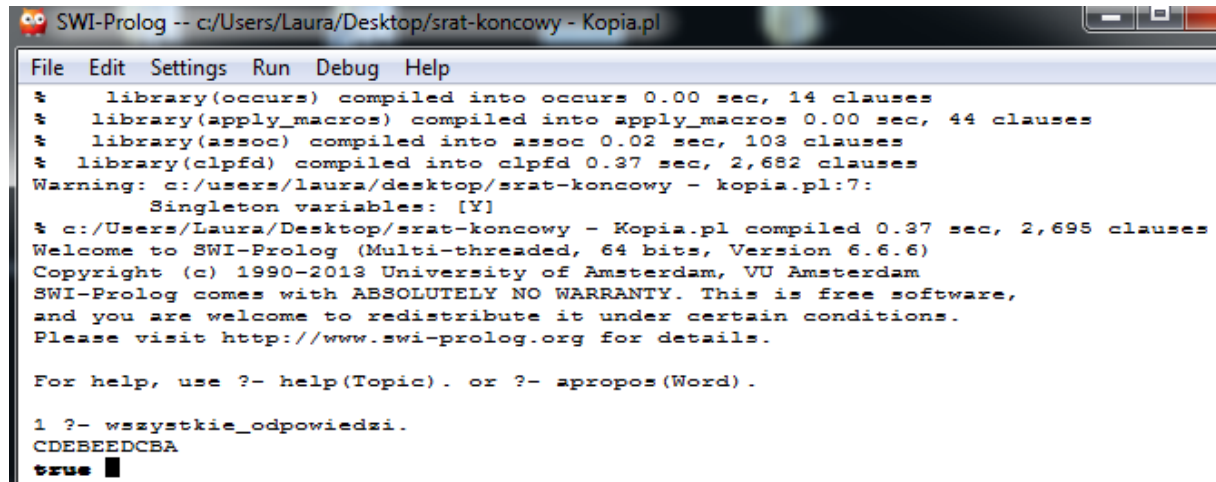
## **V. Użytkowanie i testowanie systemu**

Problem SRAT ma tylko jeden zestaw pytań i odpowiedzi, które zawsze są takie same. Program działa prawidłowo, jeśli sam – na podstawie podanych zależności – zwróci odpowiedni zestaw odpowiedzi.

Aby wywołać program w środowisku SWI-Prolog należy wpisać:

*?- wszystkie\_odpowiedzi.*

Następnie powinien wyświetlić się zestaw odpowiedzi.



```
SWI-Prolog -- c:/Users/Laura/Desktop/srata-koncowy - Kopia.pl
File Edit Settings Run Debug Help
% library(occurs) compiled into occurs 0.00 sec, 14 clauses
% library(apply_macros) compiled into apply_macros 0.00 sec, 44 clauses
% library(assoc) compiled into assoc 0.02 sec, 103 clauses
% library(clpfd) compiled into clpfd 0.37 sec, 2,682 clauses
Warning: c:/users/laura/desktop/srata-koncowy - kopia.pl:7:
Singleton variables: [Y]
% c:/Users/Laura/Desktop/srata-koncowy - Kopia.pl compiled 0.37 sec, 2,695 clauses
Welcome to SWI-Prolog (Multi-threaded, 64 bits, Version 6.6.6)
Copyright (c) 1990-2013 University of Amsterdam, VU Amsterdam
SWI-Prolog comes with ABSOLUTELY NO WARRANTY. This is free software,
and you are welcome to redistribute it under certain conditions.
Please visit http://www.swi-prolog.org for details.

For help, use ?- help(Topic). or ?- apropos(Word).

1 ?- wszystkie_odpowiedzi.
CDEBEEDCBA
true
```

Zwrócona wartość **true** oznacza, że wszystkie pytania i funkcje zostały zdefiniowane poprawnie.

W przypadku gdyby program zwrócił wartość **false** (wtedy także nie wypisałby odpowiedzi), można przetestować definicję każdego pytania oraz funkcji, podając „na sztywno” wartości wszystkich pytań i zostawiając definicję jedynie tych struktur, które chcemy przetestować.

## VI. Tekst programu

```
:- use_module(library(clpfd)).

zlicz(_, [], 0) :- !.
zlicz(X, [X|T], N) :-
    zlicz(X, T, N1),
    N is N1 + 1.
zlicz(X, [Y|T], N) :-
    zlicz(X, T, N).

srata(Rozwiazanie) :-

    Rozwiazanie=[Q1,Q2,Q3,Q4,Q5,Q6,Q7,Q8,Q9,Q10],
    Zliczenia=[N1,N2,N3,N4,N5],
    Rozwiazanie ins 1..5,
    Zliczenia ins 0..10,

    N1 + N2 + N3 + N4 + N5 #= 10,
```

% 1. Pierwsze pytanie, którego odpowiedzią jest B, to pytanie:

$Q1 \neq 1 \iff Q1 \neq 2 \wedge Q2 \neq 2,$   
 $Q1 \neq 2 \iff Q2 \neq 2 \wedge Q3 \neq 2,$   
 $Q1 \neq 3 \iff Q1 \neq 2 \wedge Q2 \neq 2 \wedge Q3 \neq 2 \wedge Q4 \neq 2,$   
 $Q1 \neq 4 \iff Q1 \neq 2 \wedge Q2 \neq 2 \wedge Q3 \neq 2 \wedge Q4 \neq 2$   
 $\wedge Q5 \neq 2,$   
 $Q1 \neq 5 \iff Q1 \neq 2 \wedge Q2 \neq 2 \wedge Q3 \neq 2 \wedge Q4 \neq 2$   
 $\wedge Q5 \neq 2 \wedge Q6 \neq 2,$

% 2. Jedyne dwa następujące po sobie pytania z identyczną odpowiedzią to:

$Q2 \neq 1 \iff Q2 \neq Q3 \wedge Q3 \neq Q4 \wedge Q4 \neq Q5 \wedge Q5 \neq$   
 $Q6 \wedge Q6 \neq Q7,$   
 $Q2 \neq 2 \iff Q2 \neq Q3 \wedge Q3 \neq Q4 \wedge Q4 \neq Q5 \wedge Q5 \neq$   
 $Q6 \wedge Q6 \neq Q7,$   
 $Q2 \neq 3 \iff Q2 \neq Q3 \wedge Q3 \neq Q4 \wedge Q4 \neq Q5 \wedge Q5 \neq$   
 $Q6 \wedge Q6 \neq Q7,$   
 $Q2 \neq 4 \iff Q2 \neq Q3 \wedge Q3 \neq Q4 \wedge Q4 \neq Q5 \wedge Q5 \neq$   
 $Q6 \wedge Q6 \neq Q7,$   
 $Q2 \neq 5 \iff Q2 \neq Q3 \wedge Q3 \neq Q4 \wedge Q4 \neq Q5 \wedge Q5 \neq$   
 $Q6 \wedge Q6 \neq Q7,$

% 3. Odpowiedź na to pytanie jest taka sama jak odpowiedź na pytanie:

$Q1 \neq Q2, Q7 \neq Q8, Q8 \neq Q9, Q9 \neq Q10,$

$Q3 \neq 1 \iff Q1 \neq Q3,$   
 $Q3 \neq 2 \iff Q2 \neq Q3,$   
 $Q3 \neq 3 \iff Q4 \neq Q3,$   
 $Q3 \neq 4 \iff Q7 \neq Q3,$   
 $Q3 \neq 5 \iff Q6 \neq Q3,$

% 4. Liczba pytań z odpowiedzią A to:

$Q4 \neq 1 \iff N1 \neq 0,$   
 $Q4 \neq 2 \iff N1 \neq 1,$   
 $Q4 \neq 3 \iff N1 \neq 2,$   
 $Q4 \neq 4 \iff N1 \neq 3,$   
 $Q4 \neq 5 \iff N1 \neq 4,$

% 5. Odpowiedź na to pytanie jest taka sama jak odpowiedź na pytanie:

$Q5 \neq 1 \iff Q10 \neq Q5,$   
 $Q5 \neq 2 \iff Q9 \neq Q5,$

$Q5 \# = 3 \# \Leftrightarrow Q8 \# = Q5,$   
 $Q5 \# = 4 \# \Leftrightarrow Q7 \# = Q5,$   
 $Q5 \# = 5 \# \Leftrightarrow Q6 \# = Q5,$

% 6.Liczba pytań z odpowiedzią A równa się liczbie pytań z odpowiedzią:

$Q6 \# = 1 \# \Leftrightarrow N1 \# = N2,$   
 $Q6 \# = 2 \# \Leftrightarrow N1 \# = N3,$   
 $Q6 \# = 3 \# \Leftrightarrow N1 \# = N4,$   
 $Q6 \# = 4 \# \Leftrightarrow N1 \# = N5,$   
 %żadne z powyższych:  
 $Q6 \# = 5 \# \Leftrightarrow N1 \# \setminus = N2 \# / \setminus N1 \# \setminus = N3 \# / \setminus N1 \# \setminus = N4 \# / \setminus N1 \# \setminus =$   
 $N5,$

% 7.Alfabetycznie, odległość odpowiedzi na to pytanie do odpowiedzi na następne pytanie to:

$Q7 \# = 1 \# \Leftrightarrow Q8 - Q7 \# = 4,$   
 $Q7 \# = 2 \# \Leftrightarrow Q8 - Q7 \# = 3,$   
 $Q7 \# = 3 \# \Leftrightarrow Q8 - Q7 \# = 2 \# \setminus / Q7 - Q8 \# = 2,$   
 $Q7 \# = 4 \# \Leftrightarrow Q8 - Q7 \# = 1 \# \setminus / Q7 - Q8 \# = 1,$   
 $Q7 \# = 5 \# \Leftrightarrow Q8 \# = 5,$

% 8.Liczba odpowiedzi, które są samogłoskami to:

$Q8 \# = 1 \# \Leftrightarrow N1 + N5 \# = 2,$   
 $Q8 \# = 2 \# \Leftrightarrow N1 + N5 \# = 3,$   
 $Q8 \# = 3 \# \Leftrightarrow N1 + N5 \# = 4,$   
 $Q8 \# = 4 \# \Leftrightarrow N1 + N5 \# = 5,$   
 $Q8 \# = 5 \# \Leftrightarrow N1 + N5 \# = 6$

% 9.Liczba odpowiedzi, które są spółgłoskami jest:

%liczbą pierwszą:  
 $Q9 \# = 1 \# \Leftrightarrow (N2 + N3 + N4) \# = 2 \# \setminus / (N2 + N3 + N4) \# = 3 \# \setminus /$   
 $(N2 + N3 + N4) \# = 5 \# \setminus / (N2 + N3 + N4) \# = 7,$

%silnią:  
 $Q9 \# = 2 \# \Leftrightarrow (N2 + N3 + N4) \# = 1 \# \setminus / (N2 + N3 + N4) \# = 2 \# \setminus /$   
 $(N2 + N3 + N4) \# = 6,$

%sześcianiem:  
 $Q9 \# = 4 \# \Leftrightarrow (N2 + N3 + N4) \# = 0 \# \setminus / (N2 + N3 + N4) \# = 1 \# \setminus /$   
 $(N2 + N3 + N4) \# = 8,$

%kwadratem



```
Q9 #= 3 #<==> (N2 + N3 + N4) #= 0 #\ / (N2 + N3 + N4) #= 1 #\ /  
(N2 + N3 + N4) #= 4 #\ / (N2 + N3 + N4) #= 9,
```

```
%podzielna przez 5:
```

```
Q9 #= 5 #<==> (N2 + N3 + N4) mod 5 #= 0,
```

```
% 10.Odpowiedzią na pytanie jest:
```

```
label(Zliczenia),  
label(Rozwiazanie),  
zlicz(1,Rozwiazanie,N1),  
zlicz(2,Rozwiazanie,N2),  
zlicz(3,Rozwiazanie,N3),  
zlicz(4,Rozwiazanie,N4),  
zlicz(5,Rozwiazanie,N5).
```

```
wszystkie_odpowiedzi:-srat(N),  
maplist(plus(64),N,M),maplist(put_char,M).
```