

Programowanie Deklaratywne

Laboratorium nr 1

Opracowane przez:
mgr inż. Artur Samojluk

UWM w Olsztynie

Programowanie w języku logiki.

PROLOG

Wstęp do ćwiczenia

Programowanie deklaratywne - jest to rodzaj programowania należący do rodziny paradygmatów programowania. Programowanie deklaratywne z natury nie są imperatywne, oznacza to że opisujemy końcowe warunki jakie musi spełnić program, a nie zestaw kroków które do tego prowadzą.

Prolog – język programowania logicznego. Język Prolog został zaprojektowany jako język do automatycznej analizy języków naturalnych. Prolog jest językiem deklaratywnym o ogólnym zastosowaniu. Z dużym powodzeniem stosuje się go do programów związanych ze sztuczną inteligencją.

Projekt proszę wysłać na e-mail, spakowany zip'em, w katalogu:

grupa_nr_lab_imie_nazwisko_nr_indeksu (np. gr1_lab1_Jan_Kowalski_123456). Z każdej sesji prologowej dołączyć zrzut ekranu z dodanym na końcu: `print(imie_nazwisko)`. w formacie jpg lub png.

Ćwiczenie 1 – Instalacja środowiska Prolog (w przypadku zajęć zdalnych)

Link do dokumentacji -> <https://www.swi-prolog.org/> -

Zainstaluj środowisko programistyczne Prolog -> <https://www.swi-prolog.org/download/stable>

Ćwiczenie 2 – Zapoznaj się z materiałem

1. Literatura:

http://enzomind.com/files/uwm/wyklady/ProgDeklaratywne/PL01_Literatura_programowanie_w_logice_Prolog.pdf

2. Wprowadzenie:

http://enzomind.com/files/uwm/wyklady/ProgDeklaratywne/PL02_Programowanie_w_jezyku_logiki_wprowadzenie.pdf

3. Podstawy Prolog:

http://enzomind.com/files/uwm/wyklady/ProgDeklaratywne/PL03_Podstawowe_elementy_jezyka_Prolog.pdf

4. Budowa programu w Prolog:

http://enzomind.com/files/uwm/wyklady/ProgDeklaratywne/PL04_Budowa_programu_w_Prologu.pdf

Ćwiczenie 3 – Pierwszy program

1. Po uruchomieniu środowiska Prolog, otworzy się konsola.



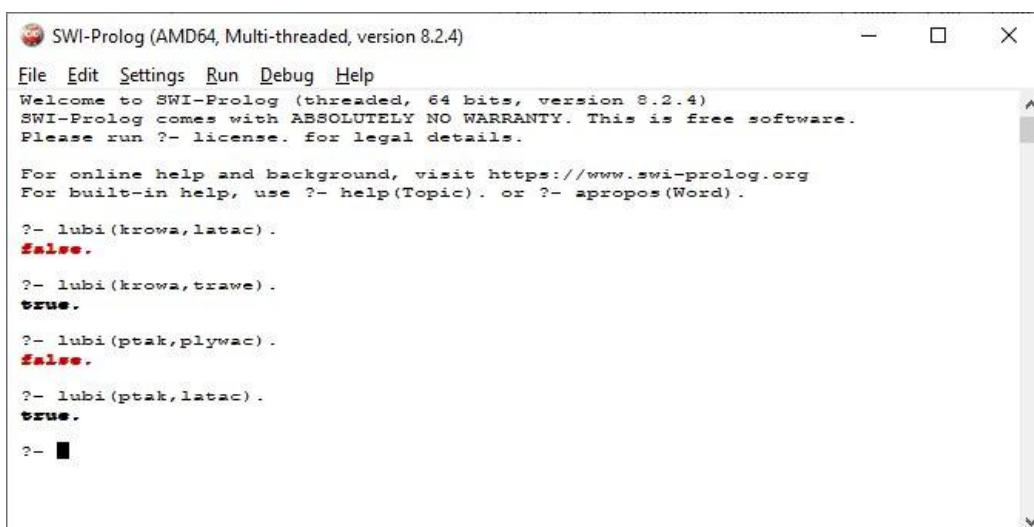
2. Na pasku opcji wybieramy **File -> New**
3. Wpisujemy nazwę projektu wraz z rozszerzeniem -> **pierwszy_program.pl** i klikamy **Zapisz**
4. Otworzy się okno edytora Prolog dla utworzonego projektu, następnie wpisujemy kod:

lubi(krowa,trawe).

lubi(ptak,latac).

lubi(ryba,plywac).

5. Następnie klikamy na pasku edytora **Compile -> Compile to buffer** (potwierdzamy zapisanie projektu). Nasz projekt jest skompilowany i możemy **prześć do konsoli**.
6. W konsoli (sesja prologowa) testujemy nasz program:



TIP 1. Jeśli istnieje więcej niż jedno rozwiązanie, wtedy wynik wyświetla się ze ; na końcu. Ostatnie rozwiązanie wyświetla się z . na końcu. Aby zobaczyć kolejne rozwiązania wciśnij TAB, gdy pojawi się ; (średnik).

Ćwiczenie 4 – Zadanie z klockami 1

a) Utwórz plik klocki1.pl i wpisz:

```
1 % PROGRAM: klocki_1
2 % Baza wiedzy o układzie klocków
3 % Definiowane predykaty:
4 % na/2
5 % =====
6
7
8 % na(X,Y)
9 % opis: spełniony, gdy klocek x leży
10 % bezpośrednio na klocku Y
11 % -----na/2
12 na(c,a).
13 na(c,b).
14 na(d,c).
15 % -----na/2
16
17 /*
18 Informacje o budowie programu:
19 Program składa się z 3 klauzul.
20 Program zawiera 1 definicję relacji.
21 Jest to relacja na/2.
22 Definicja relacji na/2 składa się z
23 3 klauzul, które są faktami.
24 */
```

b) Sesja prologowa (konsola) -> wykonaj poniższe polecenia dla programu:

```
27 Sesja prologowa
28
29 1. czy klocek d leży na klocku c?
30 ?- na(d,c).
31 true.
32
33 2. czy klocek c leży na klocku a?
34 na(c,a).
35 true.
36
37 3. czy klocek b leży na klocku c?
38 ?- na(b,c).
39 false.
40
41 4. Jaki klocek leży na klocku c?
42 ?- na(X,c).
43 X = d.
44
45 5. Na jakim klocku leży klocek c?
46 ?- na(c,X).
47 X = a ;
48 X = b.
49
50 6. Para klocków (X,Y), taka że X leży na Y?
51 ?- na(X,Y).
52 X = c,
53 Y = a ;
54 X = c,
55 Y = b ;
56 X = d,
57 Y = c.
58
59 7. czy jakiś klocek leży na c?
60 ?- na(_,c).
61 true.
62
63 8. czy jakiś klocek leży pod klockiem c?
64 ?- na(c,_).
65 true ;
66 true.
67
68 9. między jakimi klockami leży klocek c?
69 ?- na(X,c),na(c,Y).
70 X = d,
71 Y = a ;
72 X = d,
73 Y = b.
74
75 10. Jaki klocek leży między dwoma innymi klockami?
76
77 ?- na(_,X),na(X,_).
78 X = c ;
79 X = c.
```