

# Laborator 4

2017-2018

Programare Logică

# Laboratorul 4

## TODO

- Diverse exerciții.

## Exercițiul 1: drumurile într-un labirint

Baza de cunoștințe din maze.pl descrie un labirint.

Faptele indică ce puncte sunt conectate (din ce punct se poate ajunge într-un alt punct într-un pas).

Drumurile sunt cu sens unic (se poate merge pe ele doar într-o direcție).

De exemplu, se poate ajunge într-un pas de la 1 la 2, dar nu și invers.

Adăugați un predicat `path/2` care indică dacă dintr-un punct puteți să ajungeți într-un alt punct (în mai mulți pași), legând conexiunile din baza de cunoștințe.

## Exercițiul 1 (cont.)

Puneți următoarele întrebări:

- ☐ Puteți ajunge din punctul 5 în punctul 10?
- ☐ În ce puncte puteți să ajungeți plecând din 1?
- ☐ Din ce puncte puteți să ajungeți în punctul 13?

## Exercițiul 2: numere naturale

În acest exercițiu vom reprezenta numerele sub următoarea formă:

0	:	[]
1	:	[x]
2	:	[x,x]
3	:	[x,x,x]
4	:	[x,x,x,x]
...		

## Exercițiul 2 (cont.)

a) Definiți un predicat `successor/2` care întoarce în al doilea argument succesorul numărului dat ca prim argument.

### Exemplu:

```
?- successor([x,x,x],Result).  
Result = [x,x,x,x]
```

```
?- successor([],Result).  
Result = [x]
```

## Exercițiul 2 (cont.)

b) Definiți un predicat `plus/3` care adună două numere.

### **Exemplu:**

```
?- plus([x, x], [x, x, x, x], Result).  
Result = [x, x, x, x, x, x]
```

## Exercițiul 2 (cont.)

c) Definiți un predicat `times/3` care înmulțește două numere.

### **Exemplu:**

```
?- times([x, x], [x, x, x, x], Result).  
Result = [x, x, x, x, x, x, x, x]
```



## Exercițiul 3

Definiți un predicat `element_at/3` care, primind o listă și un număr natural  $n$ , întoarce al  $n$ -ulea element din listă.

### Exemplu:

```
?- element_at([tiger, dog, teddy_bear, horse, cow], 3, X).  
X = teddy_bear
```

```
?- element_at([a, b, c, d], 27, X).  
false
```

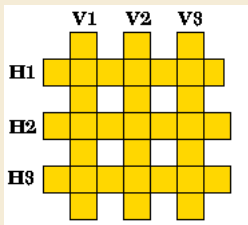
# Practică

## Exercițiul 4: cuvinte încrucișate

Sașe cuvinte din engleză

abalone, abandon, anagram, connect, elegant, enhance

trebuie aranjate într-un puzzle de cuvinte încrucișate ca în figură.



Aici găsiți o bază de cunoștințe ce reprezintă un lexicon ce conține aceste cuvinte.

## Exercițiul 4 (cont.)

Definiți un predicat `crosswd/6` care calculează toate variantele în care puteți completa grila. Primele trei argumente trebuie să fie cuvintele pe verticală, de la stânga la dreapta, (V1,V2,V3), iar următoarele trei argumente trebuie să fie cuvintele pe orizontală, de sus în jos (H1,H2,H3).

*Hint:* Specificați că V1, V2, V3, H1, H2, H3 sunt cuvinte care au anumite litere comune. Unde este cazul, folosiți variabile anonime.

## Exercițiul 5: baza de date

În acest exercițiu vom demonstra cum se poate implementa o bază de date simplă în Prolog.

Folosiți în programul vostru următoarea bază de cunoștințe:

```
born(jan, date(20,3,1977)).  
born(jeroen, date(2,2,1992)).  
born(joris, date(17,3,1995)).  
born(jelle, date(1,1,2004)).  
born(joan, date(24,12,0)).  
born(joop, date(30,4,1989)).  
born(jannecke, date(17,3,1993)).  
born(jaap, date(16,11,1995)).
```

Reprezentăm datele calendaristice ca termeni de forma  
`date(Day,Month,Year)`.

## Exercițiul 5 (cont.)

a) Scrieți un predicat `year/2` care găsește toate persoanele născute într-un anumit an.

### Exemplu:

```
?- year(1995, Person).
```

```
Person = joris
```

```
Person = jaap
```

*Hint:* Folosiți variabile anonime.

## Exercițiul 5 (cont.)

b) Scrieți un predicat `before/2` care primește două date calendaristice și care este adevărat dacă prima expresie reprezintă o dată calendaristică înaintea datei reprezentate de a doua expresie (puteți presupune că datele sunt corecte, e.g., nu puteți primi 31 Aprilie).

### Exemplu:

```
?- before(date(31,1,1990), date(7,7,1990)).  
true
```

## Exercițiul 5 (cont.)

c) Scrieți un predicat `older/2` care este adevărat dacă persoana dată ca prim argument este mai în vârstă (strict) decât persoana dată ca al doilea argument.

### Exemplu:

```
?- older(jannecke,X).
```

```
X = joris
```

```
X = jelle
```

```
X = jaap
```



Pe data viitoare!