

Logičko i funkcijsko programiranje

Primer drugog kolokvijuma

Na *Desktop*-u u direktorijumu *Rad* kreirati direktorijum *ImePrezime_BrIndeksa* i unutar njega sačuvati programe koji sadrže rešenja datih zadataka. Nazivi fajlova **moraju** biti oblika *zadatak1.pl*, *zadatak2.pl*, *zadatak3.pl* i *zadatak4.pl*.

1. Написати пролошки предикат `prikazi_listu` који прихвата листу позитивних целих бројева и графички је представља звездицама, при чему је сваки број представљен у једном реду и једнак је броју звездица у том реду.

```
?- prikazi_listu([1,3,2,4,2]).
*
***
**
****
**
true.
```

2. Написати пролошке предикате :

- a. `skloni_duplikate` који из дате листе уклања све **суседне** дупликате.

```
?- skloni_duplikate([1, 1, 2, 2, 2, 3, 3, 4, 5, 4, 3],X).
X = [1, 2, 3, 4, 5, 4, 3] .
```

- b. `spakuj_duplikate` који за дату листу бројева формира листу у којој су сви суседни дупликати спаковани у посебну листу (подлисту). Елементи који немају дупликате, пакују се као листа од једног елемента.

```
?- spakuj_duplikate([1,2,2,2,3,3,4,5,6,6,7],X).
X = [[1], [2, 2, 2], [3, 3], [4], [5], [6, 6], [7]]
```

3. Четири студента су се такмичила у бициклизму и у пливању. Познато је да

- a. Алекса није победио ни у једном такмичењу
- b. Особа која је победила у пливању, била је трећа у бициклизму
- c. Алекса је био бољи од Јована у пливању, али је Јован био бољи од Алексе у бициклизму
- d. Јован ни у једном такмичењу није био последњи
- e. Дарко је победио у бициклизму али је Стефан био бољи од њега у пливању

Написати пролошки предикат којим се одређује редослед студената на оба такмичења.

```
?- takmicenje(Pliv,Bic).
Pliv = [stefan, aleksa, jovan, darko],
Bic = [darko, jovan, stefan, aleksa]
```

4. У поља матрице уписане су структуре облика `op(a,b)` где је `a` операција `b` цео број. Дозвољене операције су сабирање, одузимање и множење. Написати следеће пролошке предикате:

- a. `formTablu` којим се омогућава унос елемената матрице. Сваки елемент матрице је облика `op(znakOperacije,argument)`.

```
?- formTablu.
Unesi dimenzije matrice 2.
|: 2.
Unesi elemente matrice
Elementi matrice su operacije u obliku op(znak_operacije, argument)
Unesi element na poziciji 1 1
|: op(*,-1).
```

```

Unesi element na poziciji 1 2
|: op(+,10).
Unesi element na poziciji 2 1
|: op(-,3).
Unesi element na poziciji 2 2
|: op(*,5).
:- dynamic mat/3.

mat(1, 1, op(*, -1)).
mat(1, 2, op(+, 10)).
mat(2, 1, op(-, 3)).
mat(2, 2, op(*, 5)).

```

- b. `nadjiSve` : којим се проналазе суме свих могућих путева кроз матрицу од горњег левог угла до доњег десног угла, при чему је дозвољено кретање у десно и на доле, и формира се и исписује листа која садржи пронађене суме.

```

?- nadjiSve.
[50,-15]
true.

```

- c. `nadjiMinimalno` : који, на основу формиране листе свих путева, проналази пут који има најмању суму и исписује одговарајућу путању кроз матрицу.

```

?- nadjiMinimalno.
Min put ima sumu -15
[mat(1,1,op(*,-1)),mat(2,1,op(-,3)),mat(2,2,op(*,5))]
True.

```

Уграђени предикати који се могу користити :

- `member(X,L)` – True ако је X елемент листе L
- `append(A,B,C)` – True ако се спајањем листе A и листе B добија листа C
- `permutation(A,B)` – True ако је A пермутација листе B