

P1	P2	P3	Total

Test laborator - Numărul I

*Pentru rezolvarea problemelor puteți defini predicate ajutătoare, puteți folosi predicate pre-definite, dar **nu** puteți folosi meta-predicate care colectează toate soluțiile (de tipul findall, findsol, bagof, setof, etc.). Nu tratați cazuri de eroare în afara celor cerute explicit de problemă.*

(P1) [5 puncte]

Scrieți un predicat `media(ListaStudenti, Student, Media)` astfel încât:

- `ListaStudenti` este o lista cu termeni `sn(NumeStudent, nota)`; în această listă, același student poate să apară de mai multe ori;
- pentru `Student` dat, `Media` va fi media notelor studentului respectiv.

Exemplu:

```
?- media([sn(adi,10), sn(stef,9), sn(stef,6), sn(adi,5), sn(adi,6)], adi, Media)
Media=7
```

(P2) [10 puncte] Presupunem ca se dă o bază de cunoștințe care conține fapte de forma `cuvant(carte,5)`. (primul argument al predicatului `cuvant` este un cuvânt, iar al doilea argument este lungimea cuvântului respectiv). Scrieți un predicat `exista(N, MultimeC)` astfel încât pentru un număr natural `N` dat, `MultimeC` să fie o listă de cuvinte din baza de cunoștințe cu proprietatea că suma lungimilor lor este egală cu `N`.

Precizări: în baza de cunoștințe fiecare cuvânt apare o singură dată; `MultimeC` trebuie să fie o mulțime (fiecare element apare o singură dată și ordinea elementelor nu contează); problema trebuie rezolvată pentru o bază de cunoștințe arbitrară.

Exemplu:

```
cuvant(testul,6).
cuvant(usor,4).
cuvant(este,4).
cuvant(ada,3).
cuvant(carte, 5).
```

```
exista(9,MultimeC).
MultimeC=[ada,testul].
```

(P3) [15 puncte]

În acest exercițiu vi se cere să evaluați o expresie într-o stare de memorie dată. Variabilele folosite în definirea expresiilor sunt declarate cu predicatul:

```
isvar(X) :- member(X,[a,b,c,d,e,f]).
```

Fiind dată o mulțime de variabile, și o constantă **zero**, o expresie este definită recursiv astfel: orice variabilă este expresie; **zero** este expresie; dacă **E1** și **E2** sunt expresii, atunci **plus(E1,E2)** și **ori(E1,E2)** sunt expresii.

Exemple de expresii:

```
a,b, zero, plus(zero,a), ori(a,b), plus(ori(a,c),zero), ori(plus(zero,a),plus(d,zero))
```

- (a) [5 puncte] Scrieți un predicat **simplifica(E,ES)** astfel încât **E** este o expresie dată, iar **ES** este expresia în care sunt simplificate operațiile cu **zero**.

Exemplu:

```
simplifica(plus(zero,a),a).  
simplifica(ori(a,zero),zero).  
simplifica(ori(plus(zero,a),plus(d,zero)), ori(a,d)).
```

- (b) [10 puncte] Scrieți un predicat **valoare(Mem,Expr, Rez)** astfel încât:

- **Mem** este o lista de termeni de tipul **vi(a,3)** care au ca prim argument o variabilă, iar al doilea argument este valoarea variabilei respective;
- dacă **Expr** este o expresie dată, **Rez** va conține valoarea expresiei în care valorile variabilelor sunt date de **Mem**, iar valoarea lui **zero** este 0.

Exemplu:

```
valoare([vi(a,3),vi(c,-4),vi(b,2)],plus(ori(a,b),c), Rez).  
Rez = 2.  
  
valoare([vi(a,3),vi(c,-4),vi(b,2)],c,Rez).  
Rez=-4.
```

Atenție: vom presupune că expresiile sunt corecte; la punctul (b) presupunem ca toate variabilele care apar în expresie apar și în memorie; variabilele apar în memorie o singură dată.