

Praktikum nr 8

Eesmärk:

Omandada praktiline oskus predikaatide repeat, fail ja cut rakendamisel faktibaasist otsingu tegemisel.

Ülesanne 1:

Tutvuda predikaatide repeat, fail ja cut kasutusvõimalustega lihtsate näidete varal:

Olgu defineeritud faktid step/3:

```
step(a,b,1).  
step(c,b,3).  
step(a,c,2).  
step(b,c,3).
```

Näide 1 (fakti otsimine prameetri tingimuse järgi):

main:-

```
repeat,           % sügavuti otsingu kordamine  
step(From,To,3), % fakti nimi, millel tehakse otsingut  
To = c,          % tagasivõtu tingimus  
write([From,To]).
```

Märkus: programm lõpetab write käsuga alles siis, kui on leitud esimene step/3 fakt, kus teine parameeter võrdub "c" –ga.

Näide 2 (prameetri tingimust rahuldavate kõikide faktide otsimine):

main:-

```
step(From,To,X),  
X == 3,  
assert(fakt(From,To)),  
fail.
```

main.

Näide 3 (faktide otsimine rekursiooniga):

```
main([]):- !, % cut-operaator – viimasel sammul 2. alternatiiv ei lähe stack'i  
main([X|List_Tail]):-  
step(From,To,X),  
assert(value(X)),  
!, % cut-operaator – siin "fail" korral rekursiooni sammu "main(.)" ei korrata  
main(List_Tail).
```

Ülesanne 2:

- Defineerida faktid, mis kirjeldavad kabelaua võimalikke seise.
- Defineerida Prologi reeglid, mis kirjeldavad kabenuppudega tehtavaid käike.

Ruudud esitame faktidega ruut/3:

```
ruut(X,Y, Status).    % kus X, Y ∈ {E,8]
    Status = 0        % tühi
    Status = 1        % valge
    Status = 2        % must
```

NB! Valged alustavad väiksemate X-indeksitega ruutudest, mustad – suuremate X-indeksitega ruutudest st. valge nupp on ruut(1,1,1).

Mängu juhib ja käikude õigsust kontrollib predikaat arbiiter, mis kutsub välja korda mööda mängivaid programme.

Mäng lõpeb, kui ühel mängijatest ei ole enam võimalik teha käike. Võitja on programm, mis sooritas viimase käigu.

Dünaamilised faktid

```
assert(fakt(14, 56)).
```

```
retract(fakt(14, 56)).
```

```
retractall(fakt(14, A).    % kustutab kõik faktid nimega fakt, mille esimene
                          % parameeter omab väärtust 14
```

```
retractall(fakt(_, _)).    % kustutab kõik faktid fakt/2,
```

:- dynamic fakt/2, fakt2/4.