

Програмски парадигми

Домашна задача 3

Стефан Милев | 206055

Дефиниран е доменот преку предикатите `ime`, `hrana`, `hobi`, `maica`, `pozicija` и `devojka`. Позициите се дефинирани како 1, 2, 3 и 4. Дадените факти се дефинирани преку предикатите `assertion1`, `assertion2`, ..., `assertion10`. Дефиниран е помошен предикат `next_to` кој кажува кои позиции се соседни. Предикатите `position` и `position_assertion` се за дефинирање на фактите поврзани со позицијата. Во овие предикати има ризик од запаѓање во јамки, па затоа се користи аргументот `AntiLoop`. Предикатот `find_tuple` прима предикат во првиот аргумент и листа од `tuples` во вториот, и определува дали има најмалку еден `tuple` која го исполнува предикатот во првиот аргумент (како операцијата `some/any` врз низи во програмски јазици). Предикатот `assert_not_equal` е точен кога сите предадени аргументи се различни меѓусебно. Предикатот `assert_tuples_not_equal` е точен кога сите предадени `tuples` се различни меѓусебно, т.е. немаат заеднички елемент. Предикатот `define_tuple` креира `tuple`. Предикатот `define_solution` дефинира листа од `tuples` за решението, при што ги дефинира сите непополнети вредности. Предикатот `reshenie` прво прави сите `tuples` да се различни една од друга, па ги доделува нивните позиции, па ги доделува нивните факти, па сите останати вредности кои не се пополнети, ќе ги пополни со претходниот предикат. Предикатот `assert_positions` ги доделува позициите на листата од `tuples`.

Добиеното решение е следното:

```
[tuple(teo,sendvich,fotografija,crvena),  
tuple(mira,pita,krstozbori,bela),  
tuple(bruno,hamburger,pishuvanje,zolta),  
tuple(igor,pica,chitanje,sina)]
```