Министерство науки и высшего образования Российской Федерации
Федеральное государственное бюджетное образовательное учреждение высшего образования
«Московский государственный технический университет имени Н. Э. Баумана (национальный исследовательский университет)»
(МГТУ им. Н. Э. Баумана)

ФАКУЛЬТЕТ «Информатика и системы управления»

КАФЕДРА «Программное обеспечение ЭВМ и информационные технологии»

ОТЧЕТ ПО ЛАБОРАТОРНОЙ РАБОТЕ №8

по курсу «Функциональное и логическое программирование»

Студент Рунов К.А.

Группа ИУ7-64Б

Оценка (баллы)

Преподаватели Толпинская Н. Б., Строганов Ю. В.

2024 г.

```
domains
  name    = symbol.
  city    = symbol.
  street  = symbol.
  phone   = symbol.

  address = addr(city, street, integer, integer).

  carmodel = symbol.
  carcolor = symbol.
  carnum   = symbol.

  price = integer.

  bank = symbol.
  bankaccount = integer.
  bankamount = integer.

  /* houseid = symbol. */
  /* landid = symbol. */
  /* boatid = symbol. */

  /* proptype = (car, house, land, boat). */
  proptype = symbol.
  propid = symbol.

    ptlist = proptype*.
    plist = price*.

predicates
  nondeterm phonebook(name, phone, address).
  /* nondeterm car(name, phone, carmodel, carcolor, carnum,
     price). */
  nondeterm car(name, phone, carmodel, carcolor, propid, price).
  nondeterm is_male(name, phone).
  nondeterm is_child(name, phone, name, phone).
  /* nondeterm is_grandchild(name, phone, name, phone). */
  nondeterm ancestor(name, phone, name, phone).
  /* nondeterm investor(name, phone, bank, bankaccount). */
```

```
39    nondeterm house(name, phone, price, propid).
40    nondeterm land(name, phone, price, propid).
41    nondeterm boat(name, phone, price, propid).
42    nondeterm property(name, phone, proptype, price, propid).
43
44      nondeterm member(proptype, ptlist).
45      nondeterm member(price, plist).
46      nondeterm collect_property_types(name, phone, ptlist, ptlist).
47      nondeterm collect_property_types(name, phone, ptlist).
48
49      nondeterm collect_property_prices(name, phone, plist, plist).
50      nondeterm collect_property_prices(name, phone, plist).
51
52      nondeterm sum(plist, price, price).
53      nondeterm sum(plist, price).
54
55      nondeterm networth(name, phone, price)
56
57  clauses
58    phonebook(abobin,     ph0,  addr(abobTown, abobAv, 0, 0)).
59    phonebook(abobina,    ph1,  addr(abobTown, abobAv, 0, 0)).
60    phonebook(abobin,     ph2,  addr(abobTown, abobAv, 1, 1)).
61    phonebook(vimina,     ph3,  addr(abobTown, abobAv, 1, 1)).
62    phonebook(abobina,    ph4,  addr(abobTown, abobAv, 2, 2)).
63    phonebook(emacsov,    ph5,  addr(abobTown, abobAv, 2, 2)).
64    phonebook(abobin,     ph6,  addr(abobTown, abobAv, 3, 3)).
65    phonebook(vSCodina,   ph7,  addr(abobTown, abobAv, 3, 3)).
66    phonebook(abobin,     ph8,  addr(abobTown, abobAv, 4, 4)).
67    phonebook(rustova,    ph9,  addr(abobTown, abobAv, 4, 4)).
68    phonebook(abobina,    ph12, addr(abobTown, abobAv, 4, 4)).
69    phonebook(abobina,    ph10, addr(abobTown, abobAv, 5, 5)).
70    phonebook(haskellov,  ph11, addr(abobTown, abobAv, 5, 5)).
71    phonebook(pupkin,     ph13, addr(pupkinTown, pupkinSt, 0, 0)).
72    phonebook(pupkina,    ph14, addr(pupkinTown, pupkinSt, 0, 0)).
73    phonebook(pupkin,     ph15, addr(pupkinTown, pupkinSt, 1, 1)).
74    phonebook(a,          ph16, addr(pupkinTown, pupkinSt, 1, 1)).
75    phonebook(pupkin,     ph17, addr(pupkinTown, pupkinSt, 2, 2)).
76    phonebook(b,          ph18, addr(pupkinTown, pupkinSt, 2, 2)).
77    phonebook(pupkin,     ph19, addr(pupkinTown, pupkinSt, 3, 3)).
78    phonebook(c,          ph20, addr(pupkinTown, pupkinSt, 3, 3)).
79    phonebook(pupkin,     ph21, addr(pupkinTown, pupkinSt, 3, 3)).
```

```
80  phonebook(emacsov,    ph22, addr(emacsTown, emacsSt, 0, 0)).
81  phonebook(lispova,    ph23, addr(emacsTown, emacsSt, 0, 0)).
82  phonebook(emacsova,   ph24, addr(emacsTown, emacsSt, 0, 0)).
83
84  is_male(abobin,     ph0).
85  is_male(abobin,     ph2).
86  is_male(emacsov,    ph5).
87  is_male(abobin,     ph6).
88  is_male(abobin,     ph8).
89  is_male(haskellov,  ph11).
90  is_male(pupkin,     ph13).
91  is_male(pupkin,     ph15).
92  is_male(pupkin,     ph17).
93  is_male(pupkin,     ph19).
94  is_male(pupkin,     ph21).
95  is_male(emacsov,    ph22).
96
97  car(abobin,     ph0,  audi,    black,  num000ber, 4091).
98  car(abobina,    ph1,  audi,    red,    num001ber, 4091).
99  car(abobin,     ph2,  audi,    black,  num002ber, 4091).
100 car(vimina,     ph3,  volvo,   green,  num003ber, 50150).
101 car(abobina,    ph4,  audi,    red,    num004ber, 4091).
102 car(emacsov,    ph5,  bmw,     purple, num005ber, 896).
103 car(abobin,     ph6,  audi,    black,  num006ber, 4091).
104 car(vSCodina,   ph7,  tesla,   blue,   num007ber, 73814).
105 car(abobin,     ph8,  audi,    black,  num008ber, 4091).
106 car(rustova,    ph9,  toyota,  orange, num009ber, 702074).
107 car(abobina,    ph12, audi,    red,    num012ber, 4091).
108 car(abobina,    ph10, audi,    red,    num010ber, 4091).
109 car(haskellov,  ph11, legs,    purple, num011ber, 1365).
110 car(pupkin,     ph13, pupcar,  pink,   num013ber, 929).
111 car(pupkina,    ph14, pupcar,  pink,   num014ber, 929).
112 car(pupkin,     ph15, pupcar,  pink,   num015ber, 929).
113 car(a,          ph16, acar,    acolor, num016ber, 1111).
114 car(pupkin,     ph17, pupcar,  pink,   num017ber, 929).
115 car(b,          ph18, bcar,    bcolor, num018ber, 2222).
116 car(pupkin,     ph19, pupcar,  pink,   num019ber, 929).
117 car(c,          ph20, ccar,    ccolor, num020ber, 3333).
118 car(pupkin,     ph21, pupcar,  pink,   num021ber, 929).
119 car(emacsov,    ph22, bmw,     purple, num022ber, 896).
120 car(lispova,    ph23, subaru,  lime,   num023ber, 508460).
```

```
121 car(emacsova, ph24, bmw, purple, num024ber, 896).
122
123 house(abobin, ph0, 1, id1).
124 house(abobina, ph1, 1, id2).
125 house(abobin, ph2, 1, id3).
126 house(vimina, ph3, 0, id4).
127 house(abobina, ph4, 1, id5).
128 house(emacsov, ph5, 6, id6).
129 house(abobin, ph6, 1, id7).
130 house(vSCodina, ph7, 4, id8).
131 house(abobin, ph8, 1, id9).
132 house(rustova, ph9, 4, id10).
133
134 land(abobina, ph12, 1, id11).
135 land(abobina, ph10, 1, id12).
136 land(haskellov, ph11, 5, id13).
137 land(pupkin, ph13, 9, id14).
138 land(pupkina, ph14, 9, id15).
139 land(pupkin, ph15, 9, id16).
140
141 boat(a, ph16, 1, id17).
142 boat(pupkin, ph17, 9, id18).
143 boat(b, ph18, 2, id19).
144 boat(pupkin, ph19, 9, id20).
145 boat(c, ph20, 3, id21).
146 boat(pupkin, ph21, 9, id22).
147 boat(emacsov, ph22, 6, id23).
148 boat(lispova, ph23, 0, id24).
149 boat(emacsova, ph24, 6, id25).
150
151 is_child(abobin, ph2, abobin, ph0).
152 is_child(abobin, ph2, abobina, ph1).
153 is_child(abobina, ph4, abobin, ph0).
154 is_child(abobina, ph4, abobina, ph1).
155 is_child(abobin, ph6, abobin, ph0).
156 is_child(abobin, ph6, abobina, ph1).
157 is_child(abobin, ph8, abobin, ph2).
158 is_child(abobin, ph8, vimina, ph3).
159 is_child(abobina, ph10, abobin, ph2).
160 is_child(abobina, ph10, vimina, ph3).
161 is_child(abobina, ph12, abobin, ph8).
```

```
162  is_child(abobina,  ph12, rustova, ph9).
163  is_child(pupkin,   ph15, pupkin,  ph13).
164  is_child(pupkin,   ph15, pupkina, ph14).
165  is_child(pupkin,   ph17, pupkin,  ph15).
166  is_child(pupkin,   ph17, a,       ph16).
167  is_child(pupkin,   ph19, pupkin,  ph17).
168  is_child(pupkin,   ph19, b,       ph18).
169  is_child(pupkin,   ph21, pupkin,  ph19).
170  is_child(pupkin,   ph21, c,       ph20).
171  is_child(emacsov,  ph22, emacsov, ph5).
172  is_child(emacsov,  ph22, abobina, ph4).
173  is_child(emacsova, ph24, emacsov, ph22).
174  is_child(emacsova, ph24, lispova, ph23).
175
176  ancestor(X, Xp, Y, Yp)
177    :- is_child(Y, Yp, X, Xp).
178  ancestor(X, Xp, Y, Yp)
179    :- is_child(Z, Zp, X, Xp), ancestor(Z, Zp, Y, Yp).
180
181  /* is_grandchild(Gc, Gcp, Gp, Gpp) */
182  /*   :- is_child(Gc, Gcp, P, Pp), is_child(P, Pp, Gp, Gpp). */
183
184  property(Name, Phone, car, Price, Id)
185    :- car(Name, Phone, _, _, Id, Price).
186  property(Name, Phone, house, Price, Id)
187    :- house(Name, Phone, Price, Id).
188  property(Name, Phone, land, Price, Id)
189    :- land(Name, Phone, Price, Id).
190  property(Name, Phone, boat, Price, Id)
191    :- boat(Name, Phone, Price, Id).
192
193    member(X, [X | _]).
194    member(X, [_ | T]) :- member(X, T).
195
196    collect_property_types(Name, Phone, Acc, PropTypes)
197        :- property(Name, Phone, PT, _, _),
198           not(member(PT, Acc)), !,
199           collect_property_types(Name, Phone, [PT|Acc],
                 PropTypes).
200    collect_property_types(_, _, PropTypes, PropTypes).
201    collect_property_types(Name, Phone, PropTypes)
```

```
202         :- collect_property_types(Name, Phone, [], PropTypes).
203
204     collect_property_prices(Name, Phone, Acc, Prices)
205         :- property(Name, Phone, _, P, _),
206             not(member(P, Acc)), !,
207             collect_property_prices(Name, Phone, [P|Acc], Prices).
208     collect_property_prices(_, _, Prices, Prices).
209     collect_property_prices(Name, Phone, Prices)
210         :- collect_property_prices(Name, Phone, [], Prices).
211
212     sum([], Acc, Acc).
213     sum([H|T], Acc, Res) :- Acc1 = Acc + H, sum(T, Acc1, Res).
214     sum(Lst, Res) :- sum(Lst, 0, Res).
215
216     networth(Name, Phone, NW)
217         :- collect_property_prices(Name, Phone, Prices),
218             sum(Prices, NW).
219
220 goal
221   /* car(Grandchild, GrandchildPhone, Model, Color, Number,
        Price), */
222   /* is_grandchild(Grandchild, GrandchildPhone, abobina, ph1), */
223   /* is_male(Grandchild, GrandchildPhone). */
224   /* property(abobin, ph2, Prop, Price, _). */
225     networth(abobin, ph2, X).
```