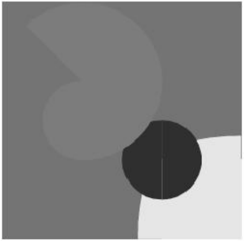
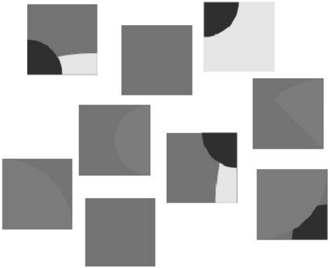
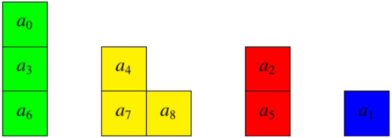
Modern art accident



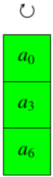


Example (Rotating polyominoes)

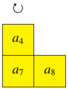
This puzzle uses one monomino, one domino, and two trominoes, for a total of nine squares. Now, assume that you can rotate the shapes. Group the four shapes in a 3×3 grid. How many solutions are there?



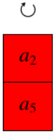
a_0	a_1	a_2
a_3	a_4	a_5
a_6	a_7	a_8



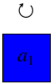
s_1



s_2

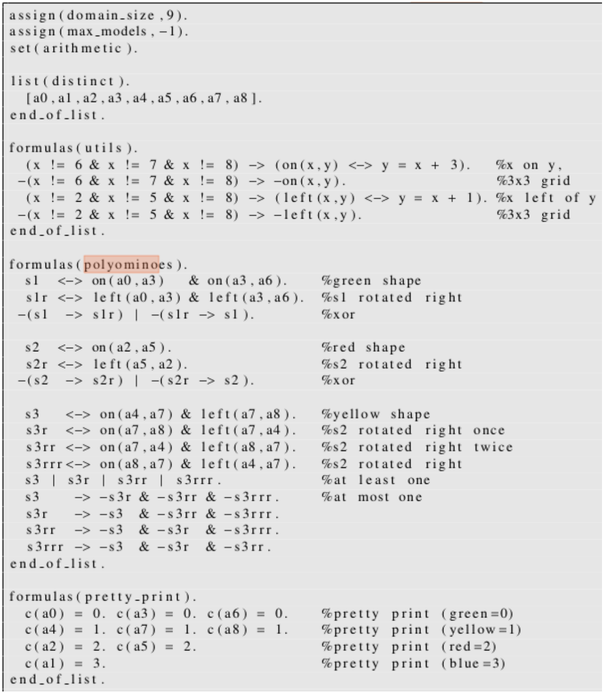


s_3



s_4

a_0	a_1	a_2
a_3	a_4	a_5
a_6	a_7	a_8



```
assign(domain_size,9).
assign(max_models,-1).
set(arithmetic).

list(distinct).
  [a0,a1,a2,a3,a4,a5,a6,a7,a8].
end_of_list.

formulas(utils).
  (x != 6 & x != 7 & x != 8) -> (on(x,y) <-> y = x + 3).   %x on y,
 -(x != 6 & x != 7 & x != 8) -> -on(x,y).                  %3x3 grid
  (x != 2 & x != 5 & x != 8) -> (left(x,y) <-> y = x + 1). %x left of y
 -(x != 2 & x != 5 & x != 8) -> -left(x,y).                %3x3 grid
end_of_list.

formulas(polyominoes).
  s1  <-> on(a0,a3)   & on(a3,a6).      %green shape
  s1r <-> left(a0,a3) & left(a3,a6).    %s1 rotated right
 -(s1  -> s1r) | -(s1r -> s1).          %xor

  s2  <-> on(a2,a5).                    %red shape
  s2r <-> left(a5,a2).                  %s2 rotated right
 -(s2  -> s2r) | -(s2r -> s2).          %xor

  s3    <-> on(a4,a7) & left(a7,a8).    %yellow shape
  s3r   <-> on(a7,a8) & left(a7,a4).    %s2 rotated right once
  s3rr  <-> on(a7,a4) & left(a8,a7).    %s2 rotated right twice
  s3rrr<-> on(a8,a7) & left(a4,a7).     %s2 rotated right
  s3 | s3r | s3rr | s3rrr.              %at least one
  s3     -> -s3r & -s3rr & -s3rrr.      %at most one
  s3r    -> -s3  & -s3rr & -s3rrr.
  s3rr   -> -s3  & -s3r  & -s3rrr.
  s3rrr  -> -s3  & -s3r  & -s3rr.
end_of_list.

formulas(pretty_print).
  c(a0) = 0. c(a3) = 0. c(a6) = 0.      %pretty print (green=0)
  c(a4) = 1. c(a7) = 1. c(a8) = 1.      %pretty print (yellow=1)
  c(a2) = 2. c(a5) = 2.                 %pretty print (red=2)
  c(a1) = 3.                            %pretty print (blue=3)
end_of_list.
```