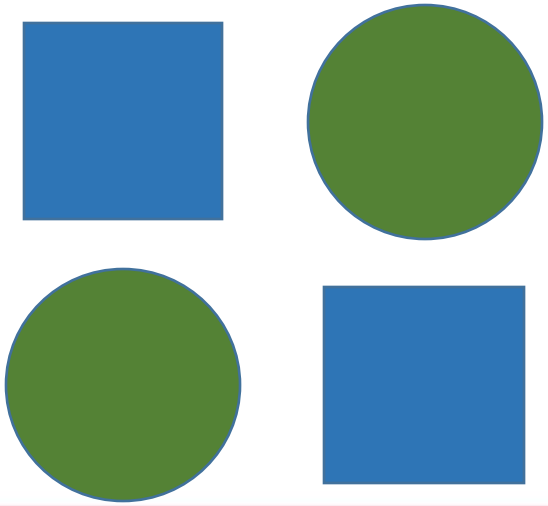
Device placement

Consider an environment characterized by a **2x2 grid to be covered by two kind of devices (A, B)**. The grid is initially empty and the devices can be placed in the grid one at a time if the following constraints are satisfied. **Type A devices can be placed in the grid only if the chosen cell is not occupied by another type A device. If there is already a type B device the type A device can be placed on top of the type B device**. A type B device can only be placed in a free cell. It is **forbidden to have two devices of the same type on a row and on a column**. The goal is to cover all the cells of the grid with devices.



Device placement

Domain

```
(define (domain device-domain)
  (:requirements :strips)
  (:predicates (at ?x ?y) (full ?x) (adj ?x ?y))

  (:action PlaceA
    :parameters (?q ?col ?row)
    :precondition (and (not (= ?col ?row)) (adj ?q ?row) (adj ?q ?col)
                       (not (at DEVA ?q))  (not (at DEVA ?row)) (not (at DEVA ?col)))
    :effect (and (at DEVA ?q) (full ?q))
  )
  (:action PlaceB
    :parameters (?q ?col ?row)
    :precondition (and (not (= ?col ?row)) (adj ?q ?row) (adj ?q ?col)
          (not (at DEVA ?q)) (not (at DEVB ?q)) (not (at DEVB ?row)) (not (at DEVB ?col)))
    :effect (and (at DEVB ?q) (full ?q))
  )
)
```

Device placement

Problem

```
(define (problem device-problem)
  (:domain device-domain)
  (:objects DEVA DEVB q1 q2 q3 q4)
  (:init
    (adj q1 q2) (adj q1 q3) (adj q2 q1) (adj q2 q4)
    (adj q3 q1) (adj q3 q4) (adj q4 q3) (adj q4 q2)
  )
  (:goal (and (full q1) (full q2) (full q3) (full q4)))
)
```