; Modelling the Wumpus World in PDDL: 1st try...
; by: Patrik Haslum
; Source web page:
;
(define (domain wumpus-a)
  (:requirements :strips) ;; maybe not necessary
  (:predicates
    (adj ?square-1 ?square-2)
    (pit ?square)
    (at ?what ?square)
    (have ?who ?what)
    (dead ?who))

  (:action move 
    :parameters (?who ?from ?to)
    :precondition (and (adj ?from ?to)
                        (not (pit ?to))
                        (at ?who ?from))
    :effect (and (not (at ?who ?from))
                  (at ?who ?to))
  )

  (:action take 
    :parameters (?who ?what ?where)
    :precondition (and (at ?who ?where)
                        (at ?what ?where))
    :effect (and (have ?who ?what)
                  (not (at ?what ?where)))
  )

  (:action shoot 
    :precondition (and (have ?who ?arrow)
                        (at ?who ?where)
                        (at ?victim ?where-victim)
                        (adj ?where ?where-victim))
    :effect (and (dead ?victim)
                  (not (at ?victim ?where-victim))
                  (not (have ?who ?arrow)))
  )
)
(define (problem wumpus-a-1)
  (:domain wumpus-a)
  (:objects
    sq-1-1 sq-1-2 sq-1-3
    sq-2-1 sq-2-2 sq-2-3
    the-gold
    the-arrow
    agent
    wumpus)
  (:init
    (adj sq-1-1 sq-1-2) (adj sq-1-2 sq-1-1)
    (adj sq-1-2 sq-1-3) (adj sq-1-3 sq-1-2)
    (adj sq-2-1 sq-2-2) (adj sq-2-2 sq-2-1)
    (adj sq-2-2 sq-2-3) (adj sq-2-3 sq-2-2)
    (adj sq-1-1 sq-2-1) (adj sq-2-1 sq-1-1)
    (adj sq-1-2 sq-2-2) (adj sq-2-2 sq-1-2)
    (adj sq-1-3 sq-2-3) (adj sq-2-3 sq-1-3)
    (pit sq-1-2)
    (at the-gold sq-1-3)
    (at agent sq-1-1)
    (have agent the-arrow)
    (at wumpus sq-2-3))
  (:goal (and (have agent the-gold) (at agent sq-1-1)))
)

Resulting plan:

(MOVE THE-GOLD SQ-1-3 SQ-2-3)
(MOVE THE-GOLD SQ-2-3 SQ-2-2)
(MOVE THE-GOLD SQ-2-2 SQ-2-1)
(MOVE THE-GOLD SQ-2-1 SQ-1-1)
(TAKE AGENT THE-GOLD SQ-1-1)