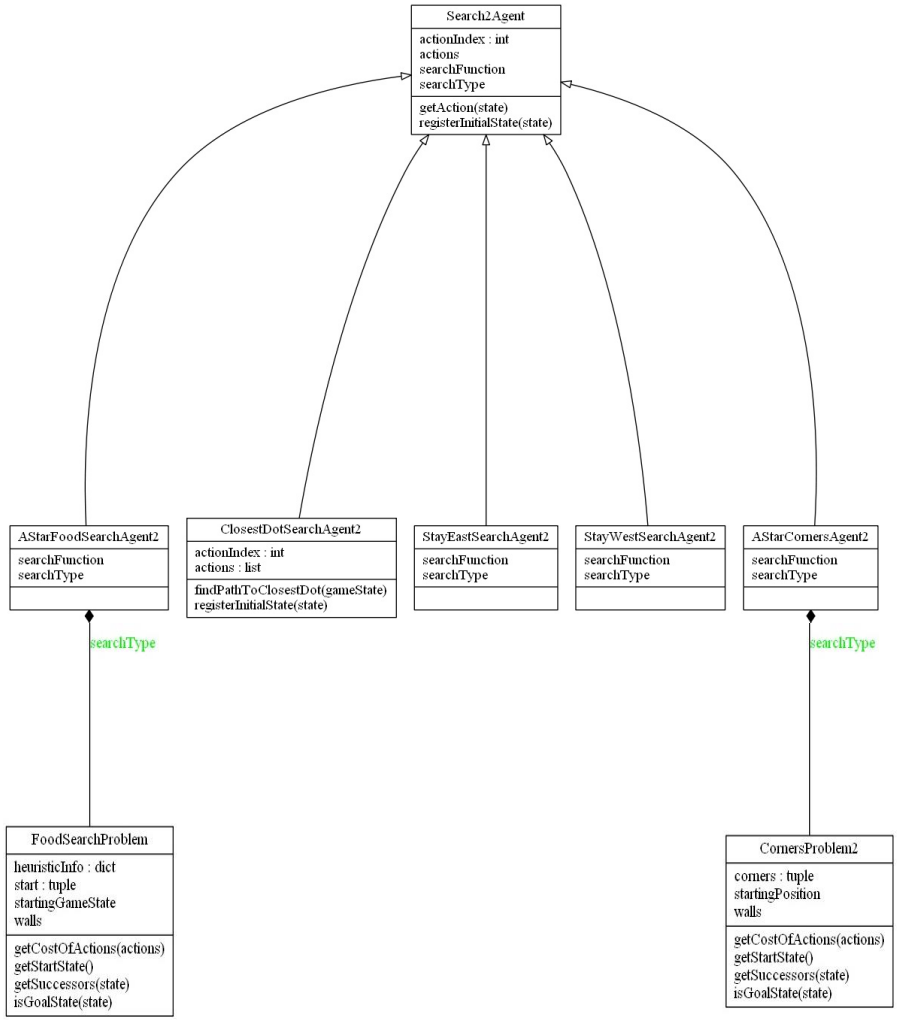
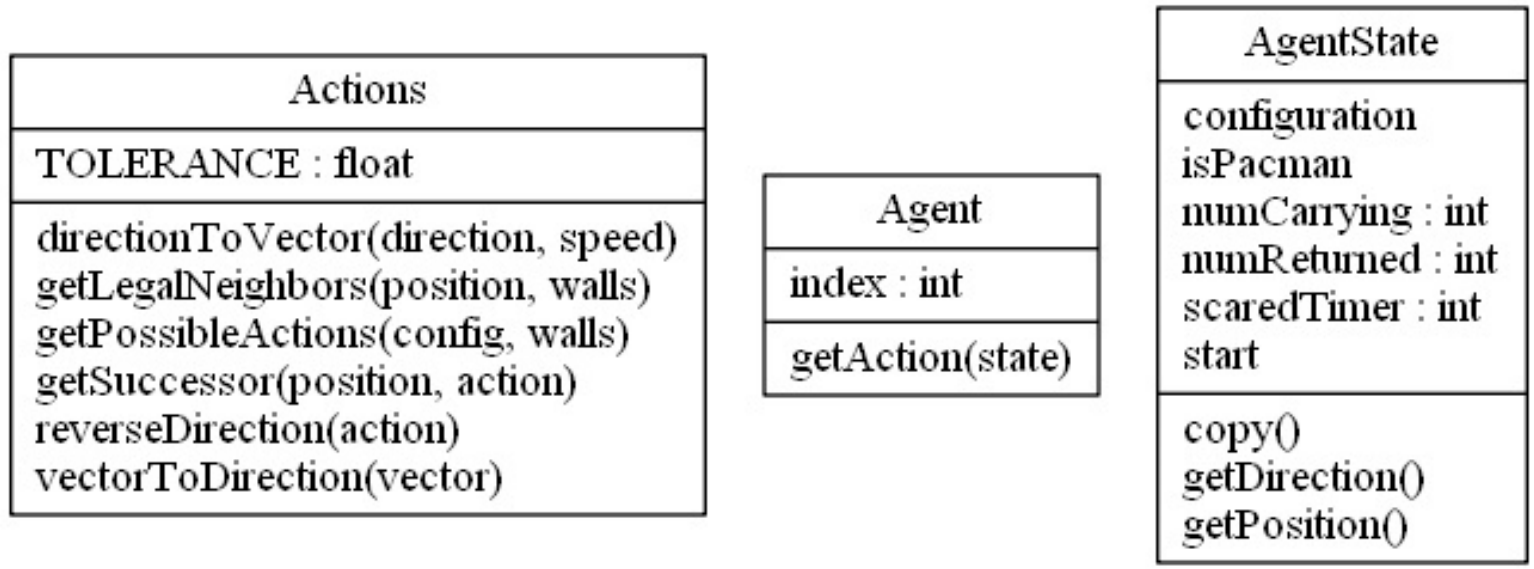
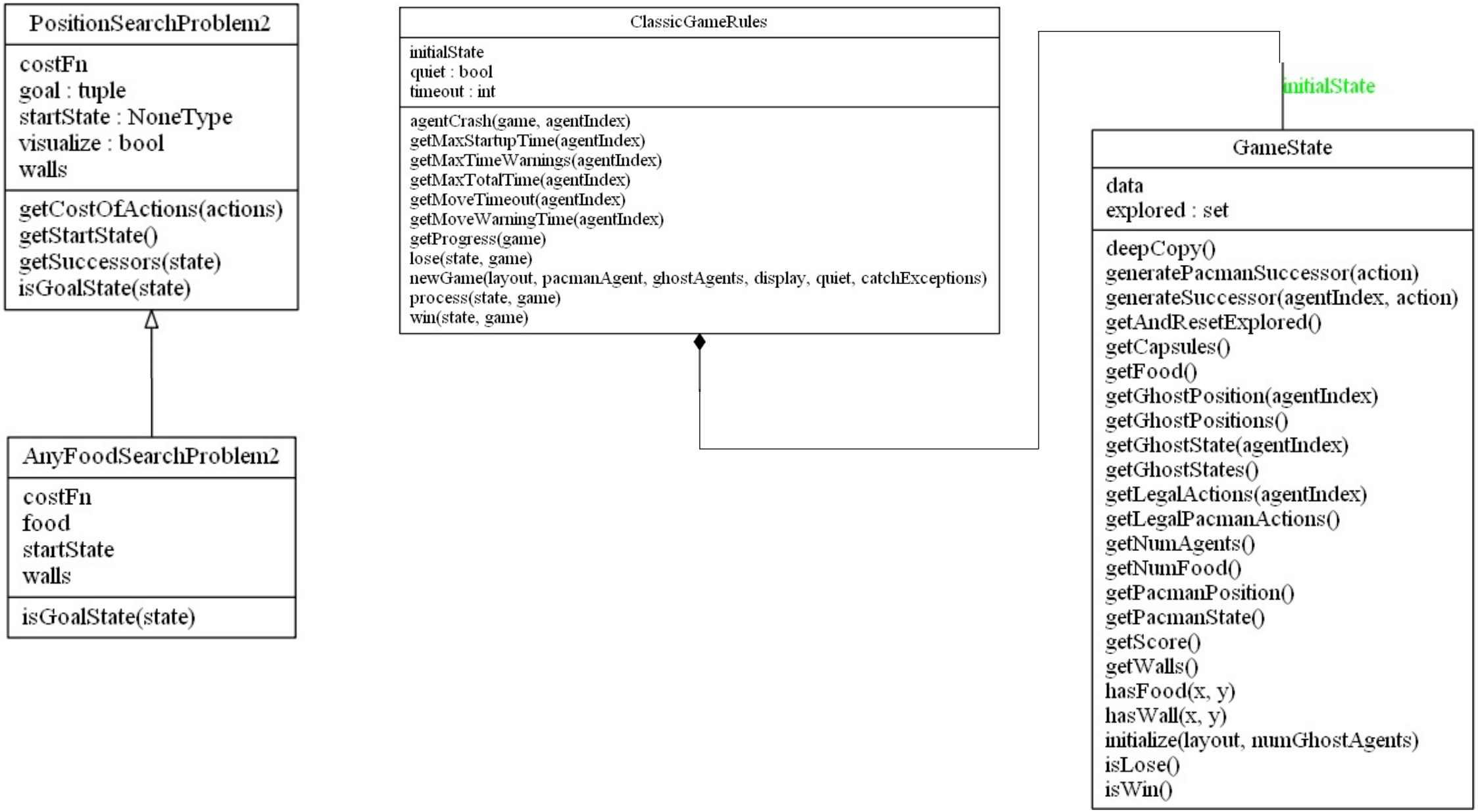
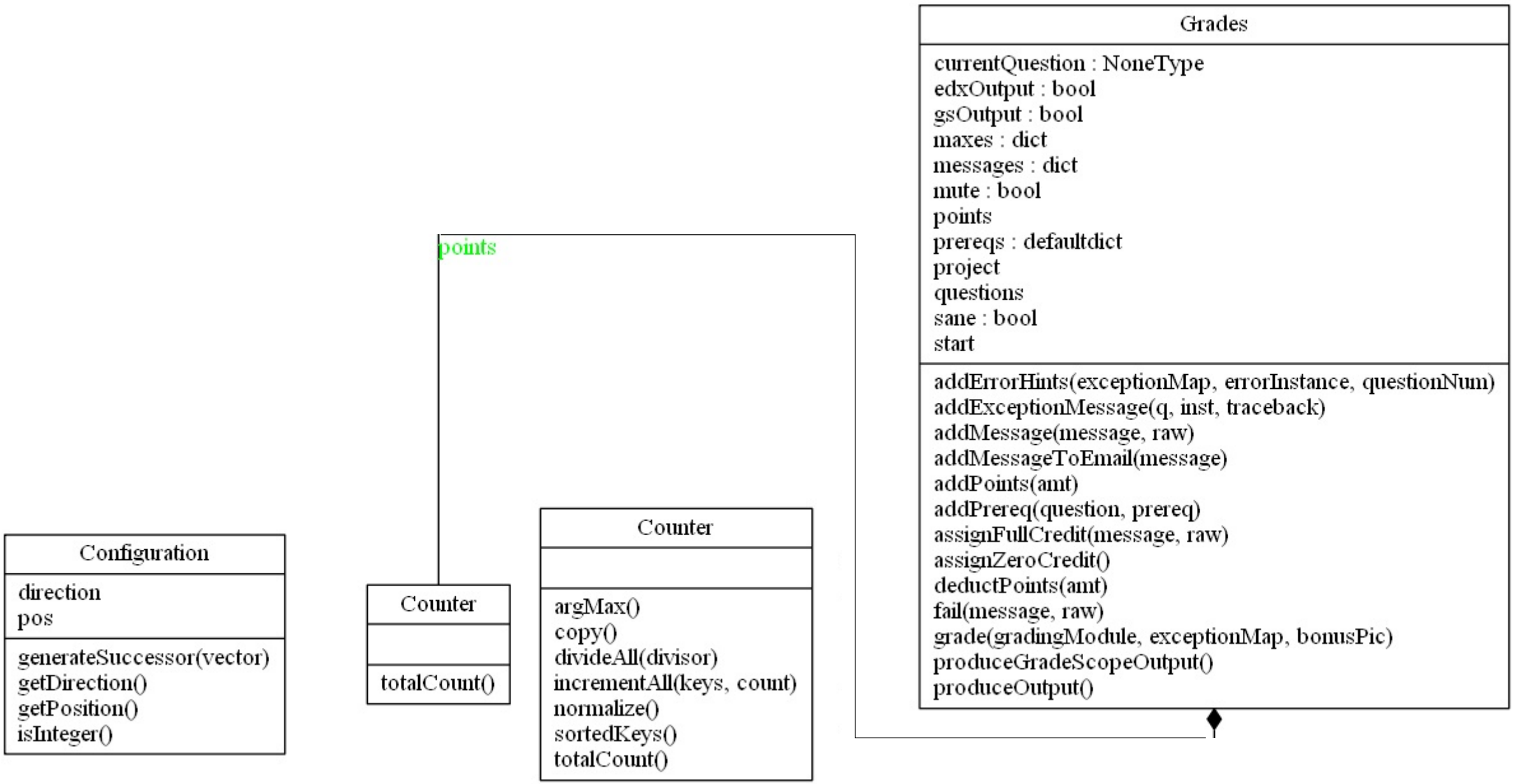
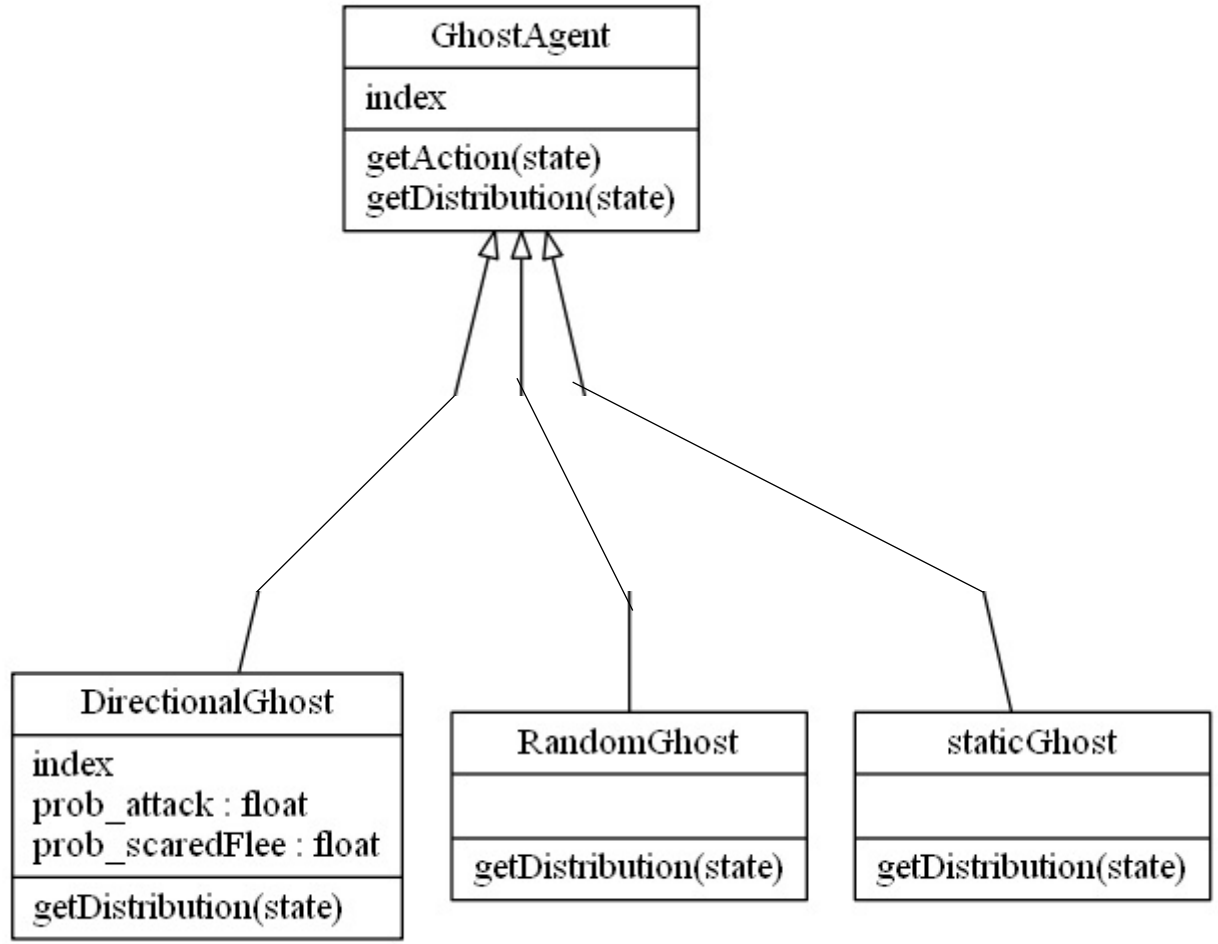
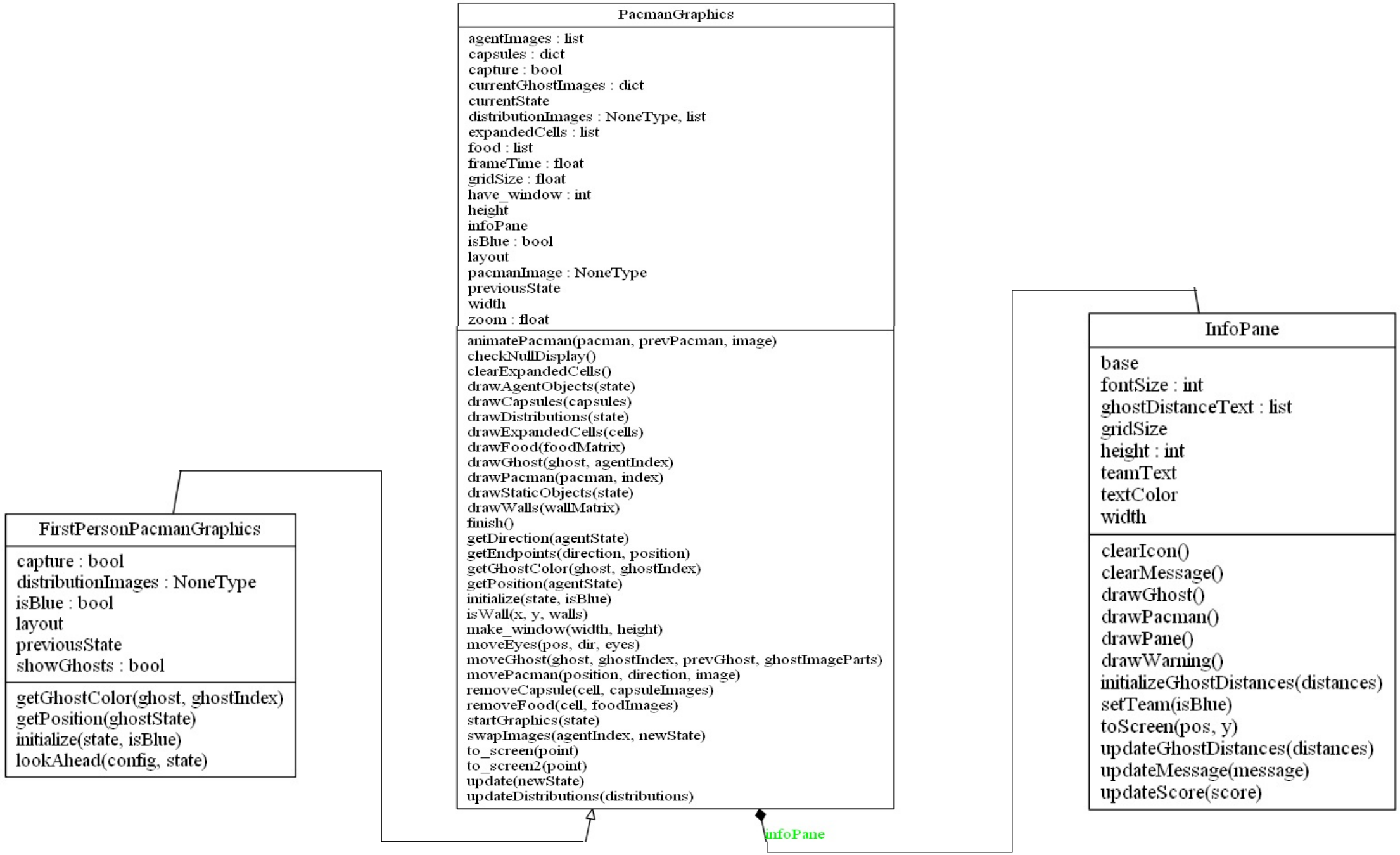

Search2Agent
actionIndex : int
actions
searchFunction
searchType
getAction(state)
registerInitialState(state)
AStarFoodSearchAgent2
searchFunction
searchType
ClosestDotSearchAgent2
actionIndex : int
actions : list
findPathToClosestDot(gameState)
registerInitialState(state)
StayEastSearchAgent2
searchFunction
searchType
StayWestSearchAgent2
searchFunction
searchType
AStarCornersAgent2
searchFunction
searchType
searchType
searchType
FoodSearchProblem
heuristicInfo : dict
start : tuple
startingGameState
walls
getCostOfActions(actions)
getStartState()
getSuccessors(state)
isGoalState(state)
CornersProblem2
corners : tuple
startingPosition
walls
getCostOfActions(actions)
getStartState()
getSuccessors(state)
isGoalState(state)

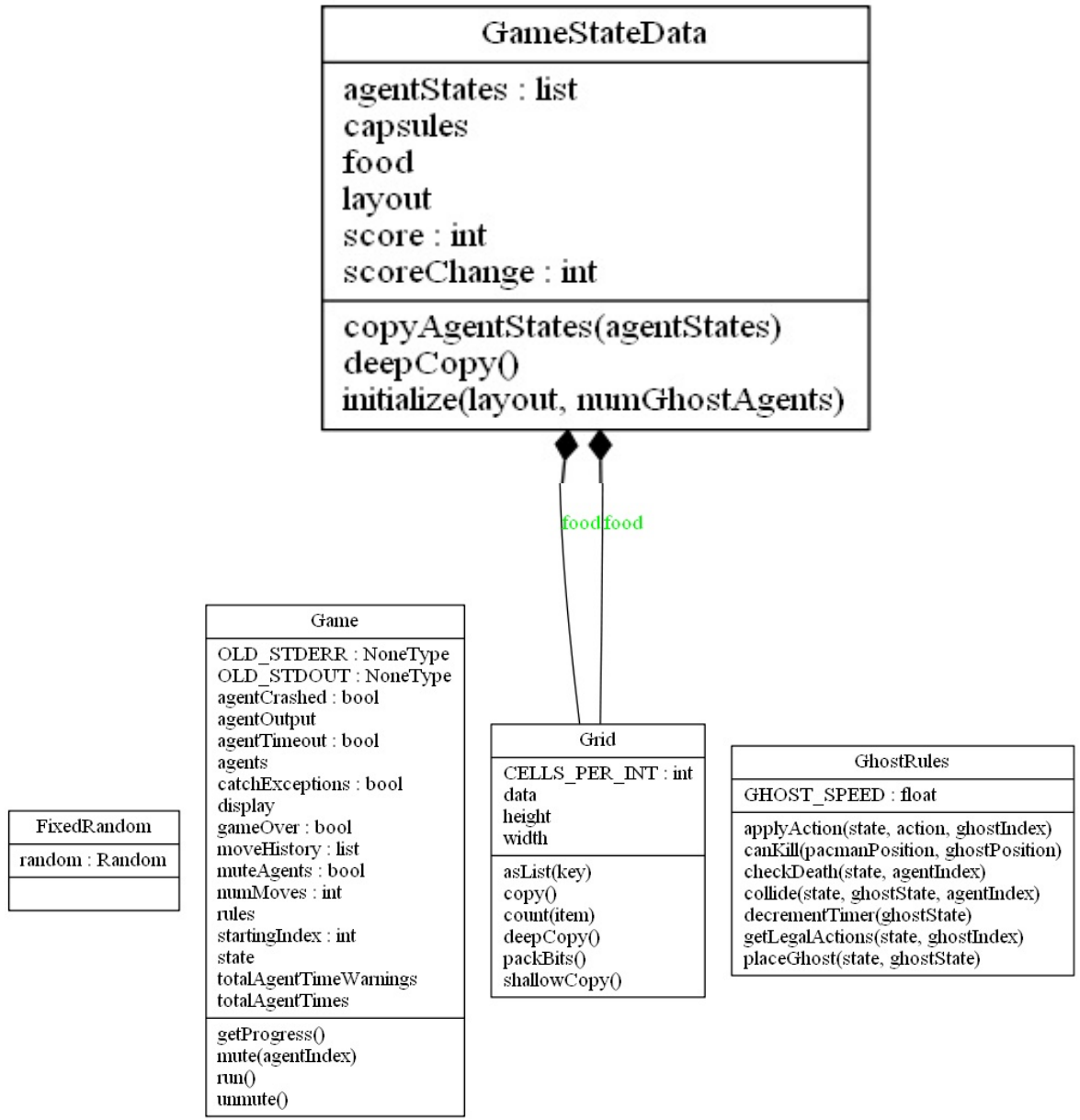

Actions
TOLERANCE : float
directionToVector(direction, speed)
getLegalNeighbors(position, walls)
getPossibleActions(config, walls)
getSuccessor(position, action)
reverseDirection(action)
vectorToDirection(vector)
Agent
index : int
getAction(state)
AgentState
configuration
isPacman
numCarrying : int
numReturned : int
scaredTimer : int
start
copy()
getDirection()
getPosition()

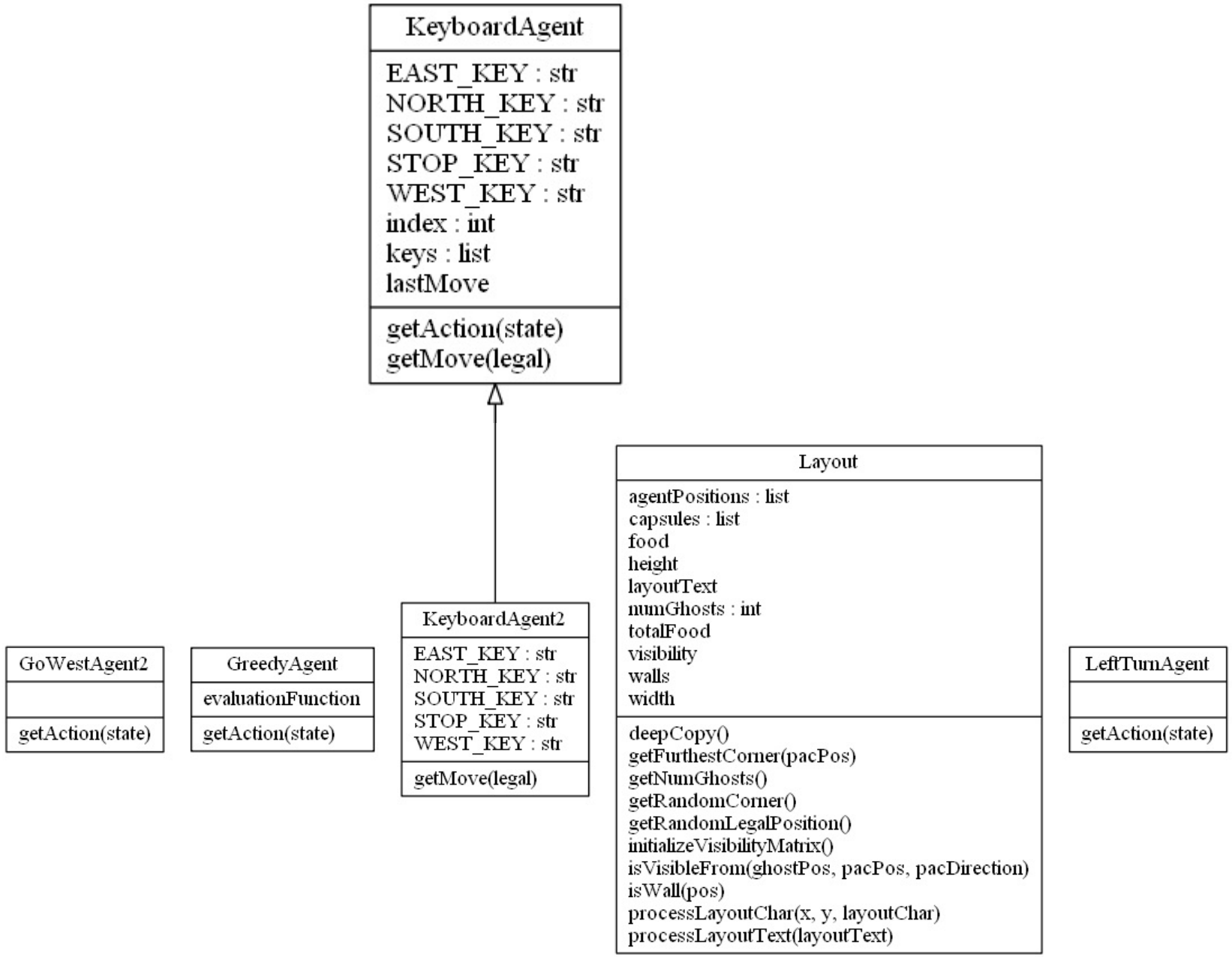

PositionSearchProblem2
costFn
goal : tuple
startState : NoneType
visualize : bool
walls
getCostOfActions(actions)
getStartState()
getSuccessors(state)
isGoalState(state)
AnyFoodSearchProblem2
costFn
food
startState
walls
isGoalState(state)
ClassicGameRules
initialState
quiet : bool
timeout : int
agentCrash(game, agentIndex)
getMaxStartupTime(agentIndex)
getMaxTimeWarnings(agentIndex)
getMaxTotalTime(agentIndex)
getMoveTimeout(agentIndex)
getMoveWarningTime(agentIndex)
getProgress(game)
lose(state, game)
newGame(layout, pacmanAgent, ghostAgents, display, quiet, catchExceptions)
process(state, game)
win(state, game)
initialState
GameState
data
explored : set
deepCopy()
generatePacmanSuccessor(action)
generateSuccessor(agentIndex, action)
getAndResetExplored()
getCapsules()
getFood()
getGhostPosition(agentIndex)
getGhostPositions()
getGhostState(agentIndex)
getGhostStates()
getLegalActions(agentIndex)
getLegalPacmanActions()
getNumAgents()
getNumFood()
getPacmanPosition()
getPacmanState()
getScore()
getWalls()
hasFood(x, y)
hasWall(x, y)
initialize(layout, numGhostAgents)
isLose()
isWin()

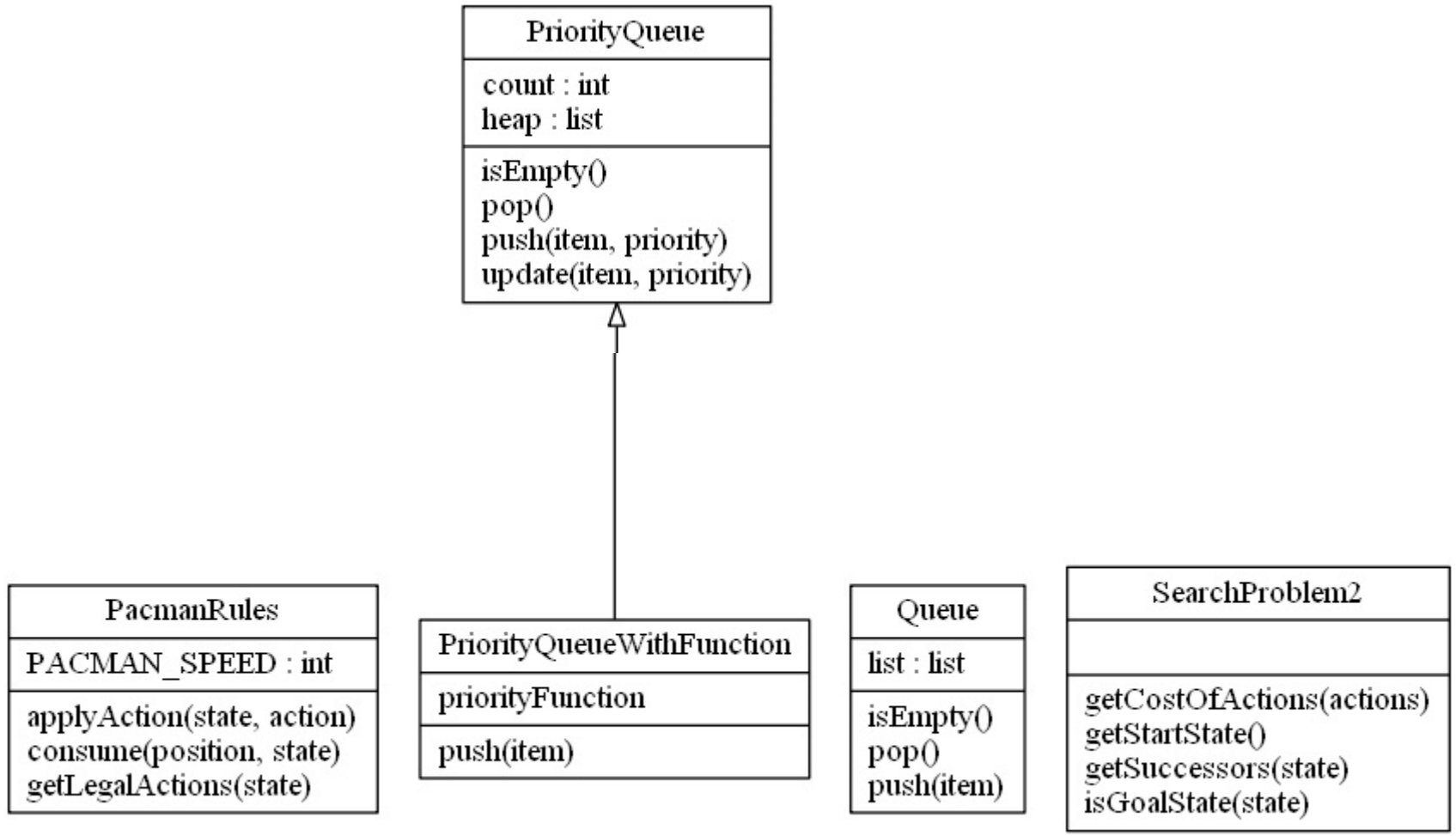

Grades
currentQuestion : NoneType
edxOutput : bool
gsOutput : bool
maxes : dict
messages : dict
mute : bool
points
prereqs : defaultdict
project
questions
sane : bool
start
addErrorHints(exceptionMap, errorInstance, questionNum)
addExceptionMessage(q, inst, traceback)
addMessage(message, raw)
addMessageToEmail(message)
addPoints(amt)
addPrereq(question, prereq)
assignFullCredit(message, raw)
assignZeroCredit()
deductPoints(amt)
fail(message, raw)
grade(gradingModule, exceptionMap, bonusPic)
produceGradeScopeOutput()
produceOutput()
points
Counter
totalCount()
Counter
argMax()
copy()
divideAll(divisor)
incrementAll(keys, count)
normalize()
sortedKeys()
totalCount()
Configuration
direction
pos
generateSuccessor(vector)
getDirection()
getPosition()
isInteger()

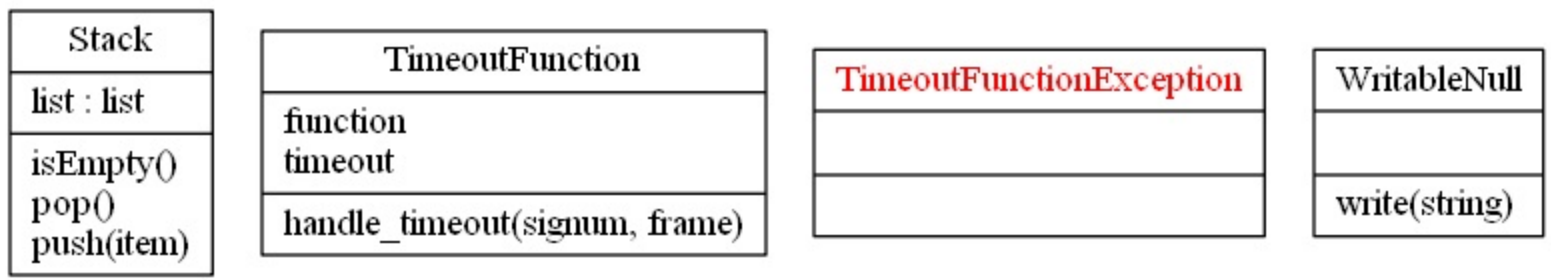

GhostAgent
index
getAction(state)
getDistribution(state)
DirectionalGhost
index
prob_attack : float
prob_scaredFlee : float
getDistribution(state)
RandomGhost
getDistribution(state)
staticGhost
getDistribution(state)


PacmanGraphics
agentImages : list
capsules : dict
capture : bool
currentGhostImages : dict
currentState
distributionImages : NoneType, list
expandedCells : list
food : list
frameTime : float
gridSize : float
have_window : int
height
infoPane
isBlue : bool
layout
pacmanImage : NoneType
previousState
width
zoom : float
animatePacman(pacman, prevPacman, image)
checkNullDisplay()
clearExpandedCells()
drawAgentObjects(state)
drawCapsules(capsules)
drawDistributions(state)
drawExpandedCells(cells)
drawFood(foodMatrix)
drawGhost(ghost, agentIndex)
drawPacman(pacman, index)
drawStaticObjects(state)
drawWalls(wallMatrix)
finish()
getDirection(agentState)
getEndpoints(direction, position)
getGhostColor(ghost, ghostIndex)
getPosition(agentState)
initialize(state, isBlue)
isWall(x, y, walls)
make_window(width, height)
moveEyes(pos, dir, eyes)
moveGhost(ghost, ghostIndex, prevGhost, ghostImageParts)
movePacman(position, direction, image)
removeCapsule(cell, capsuleImages)
removeFood(cell, foodImages)
startGraphics(state)
swapImages(agentIndex, newState)
to_screen(point)
to_screen2(point)
update(newState)
updateDistributions(distributions)
FirstPersonPacmanGraphics
capture : bool
distributionImages : NoneType
isBlue : bool
layout
previousState
showGhosts : bool
getGhostColor(ghost, ghostIndex)
getPosition(ghostState)
initialize(state, isBlue)
lookAhead(config, state)
InfoPane
base
fontSize : int
ghostDistanceText : list
gridSize
height : int
teamText
textColor
width
clearIcon()
clearMessage()
drawGhost()
drawPacman()
drawPane()
drawWarning()
initializeGhostDistances(distances)
setTeam(isBlue)
toScreen(pos, y)
updateGhostDistances(distances)
updateMessage(message)
updateScore(score)
infoPane


GameStateData
agentStates : list
capsules
food
layout
score : int
scoreChange : int
copyAgentStates(agentStates)
deepCopy()
initialize(layout, numGhostAgents)
food
food
Game
OLD_STDERR : NoneType
OLD_STDOUT : NoneType
agentCrashed : bool
agentOutput
agentTimeout : bool
agents
catchExceptions : bool
display
gameOver : bool
moveHistory : list
muteAgents : bool
numMoves : int
rules
startingIndex : int
state
totalAgentTimeWarnings
totalAgentTimes
getProgress()
mute(agentIndex)
run()
unmute()
Grid
CELLS_PER_INT : int
data
height
width
asList(key)
copy()
count(item)
deepCopy()
packBits()
shallowCopy()
GhostRules
GHOST_SPEED : float
applyAction(state, action, ghostIndex)
canKill(pacmanPosition, ghostPosition)
checkDeath(state, agentIndex)
collide(state, ghostState, agentIndex)
decrementTimer(ghostState)
getLegalActions(state, ghostIndex)
placeGhost(state, ghostState)
FixedRandom
random : Random


KeyboardAgent
EAST_KEY : str
NORTH_KEY : str
SOUTH_KEY : str
STOP_KEY : str
WEST_KEY : str
index : int
keys : list
lastMove
getAction(state)
getMove(legal)
Layout
agentPositions : list
capsules : list
food
height
layoutText
numGhosts : int
totalFood
visibility
walls
width
deepCopy()
getFurthestCorner(pacPos)
getNumGhosts()
getRandomCorner()
getRandomLegalPosition()
initializeVisibilityMatrix()
isVisibleFrom(ghostPos, pacPos, pacDirection)
isWall(pos)
processLayoutChar(x, y, layoutChar)
processLayoutText(layoutText)
KeyboardAgent2
EAST_KEY : str
NORTH_KEY : str
SOUTH_KEY : str
STOP_KEY : str
WEST_KEY : str
getMove(legal)
GoWestAgent2
getAction(state)
GreedyAgent
evaluationFunction
getAction(state)
LeftTurnAgent
getAction(state)


PriorityQueue
count : int
heap : list
isEmpty()
pop()
push(item, priority)
update(item, priority)
PacmanRules
PACMAN_SPEED : int
applyAction(state, action)
consume(position, state)
getLegalActions(state)
PriorityQueueWithFunction
priorityFunction
push(item)
Queue
list : list
isEmpty()
pop()
push(item)
SearchProblem2
getCostOfActions(actions)
getStartState()
getSuccessors(state)
isGoalState(state)


Stack
list : list
isEmpty()
pop()
push(item)
TimeoutFunction
function
timeout
handle_timeout(signum, frame)
TimeoutFunctionException
WritableNull
write(string)