

GameGrid

(fonctions globales lorsque la grille est construite avec makeGameGrid())

from gamegrid import *

GameGrid(nbHorzCells, nbVertCells, cellSize, color)	Génère une fenêtre de jeu possédant le nombre indiqué de cellules en horizontal et en vertical. Elles ont une taille de cellSize. Les lignes de la grille sont visibles et de couleur color
GameGrid(nbHorzCells, nbVertCells, cellSize, color, bgImagePath)	Idem, en utilisant l'image de fond bgImagePath
GameGrid(nbHorzCells, nbVertCells, cellSize, None, bgImagePath, False)	Idem, sans afficher les lignes de la grille (color=None) ni la barre de navigation (False)
act()	Méthode appelée par le système à chaque cycle de simulation après les Actor.act()
addActor(actor, location)	Ajoute l'acteur actor à la fenêtre de jeu à la position indiquée
addKeyListener(listener)	Enregistre la fonction de rappel listener pour gérer les événements clavier
addMouseListener(onMouseEvent, mask)	Enregistre le gestionnaire (fonction de rappel) onMouseEvent(actor, mouse, spot) avec un masque OR: GGMouse.IClick, GGMouse.IDClick, GGMouse.IDrag, GGMouse.IPress, GGMouse.IRelease, (analogue avec r), GGMouse.enter, GGMouse.move, GGMouse.leave
addStatusBar(height)	Ajoute une barre d'état à la fenêtre de jeu
delay(time)	Met le programme en pause pendant time millisecondes
doPause()	Met les cycles de simulation en pause
doStep()	Effectue la simulation pas à pas
doReset()	Remet tous les acteurs à leur position initiale et redémarre la simulation
doRun()	Démarre les cycles de simulation
getActors(Actor class)	Retourne sous forme de liste tous les acteurs du jeu appartenant à la classe spécifiée
getBg()	Retourne la référence à GGBackground
getBgColor()	Retourne la couleur d'arrière-fond
getKeyCode()	Retourne le code clavier de la dernière touche actionnée
getOneActorAt(location)	Retourne le premier acteur de la cellule indiquée par location (zéro s'il n'y a pas d'acteur)
getOneActor(Actor class)	Retourne le premier acteur de la classe class (zéro s'il n'y a pas d'acteur de la classe class)
getRandomEmptyLocation()	Retourne la position d'une cellule vide choisie au hasard dans la grille de jeu
getRandomLocation()	Retourne la position d'une cellule vide ou occupée choisie au hasard dans la grille de jeu
hide()	Cache la fenêtre de jeu sans la fermer
isAtBorder(location)	Retourne True si la cellule location est située au bord de la grille de jeu
isEmpty(location)	Retourne True si la cellule location est vide
isInGrid(location)	Retourne True si la cellule est située à l'intérieur de la grille de jeu
kbhit()	Retourne True si une touche a été enfoncée
toLocation(x, y)	Retourne la cellule contenant le point de coordonnées x et y (pixels)
refresh()	Rafraîchit la fenêtre de jeu

registerAct(onAct)	Enregistre la fonction de rappel onAct qui est appelée à chaque cycle de simulation
registerNavigation(started = onStart, stepped = onStep, paused = onPause, resetted = onReset, periodChanged = onPeriodChange)	Enregistre les fonctions de rappel onStart, onStep, onPause, onReset, onPeriodChange qui sont appelées lorsque la barre de navigation est visible. Il n'est pas nécessaire de toutes les spécifier.
removeActor (actor)	Supprime l'acteur actor de la fenêtre de jeu
removeActorsAt(location)	Supprime tous les acteurs situés dans la cellule location
removeAllActors()	Supprime tous les acteurs présents dans la fenêtre de jeu
reset()	Réinitialise la simulation en remplaçant les acteurs encore présents sur le jeu à position qu'ils occupaient avant le début de la partie
show()	Afficher la fenêtre de jeu
setBgColor(color)	Règle la couleur d'arrière-plan de la fenêtre de jeu
setSimulationPeriod (milisec)	Règle la période de la boucle de simulation
setStatusText(text)	Modifie le texte text de la barre d'état
setTitle(text)	Ajuste le titre affiché dans la barre de titre de la fenêtre

Actor

Actor(spritepath)	Génère un acteur associé à l'image de sprite chargée à partir du fichier de chemin spritepath
Actor(True, spritepath)	Idem. True indique que l'acteur peut subir des rotations
Actor(spritepath, nbSprites)	Génère un acteur avec nbSprites images de sprite différentes. Les noms de fichiers images seront composés à partir des sprite-path en ajoutant un nombre entre 0 et nbSprites-1 Exemple : index_0, index_1, ou fish_0.gif , fish_1.gif, ...
Actor(True, spritepath, nbSprites)	Idem, mais en permettant les rotations
act()	Méthode appelée périodiquement sur chaque acteur après le début du cycle de simulation
addActorCollisionListener(listener)	Enregistre le gestionnaire d'événements de collisions
addCollisionActor(actor)	Enregistre actor comme partenaire de collision
addMouseTouchListener (listener)	Enregistre le gestionnaire (fonction de rappel) pour l'événement MouseTouch
collide(actor1, actor2)	Fonction de rappel appelée lors d'une collision entre actor1 et actor2. Retourne le nombre de cycles à partir de la collision pendant lesquels les événements de collision entre actor1 et actor2 sont ignorés
getCollisionActors()	Retourne une liste de candidats à la collision
getDirection()	Retourne la direction du mouvement de l'acteur
getIdVisible()	Retourne l'ID du sprite actuellement affiché
getNeighbours(distance)	Retourne une liste de tous les acteurs qui sont éloignés de distance par rapport à l'acteur sur lequel la méthode est invoquée. Distance définie un cercle autour du centre de la cellule actuelle (unité: largeur de la cellule)
getNextMoveLocation (location)	Prédit les coordonnées de l'acteur après le prochain appel à move()
getX()	Retourne la coordonnée horizontale (dans le système de coordonnées grille) de la cellule occupée par l'acteur
getY()	Retourne la coordonnée verticale (dans le système de coordonnées grille) de la cellule occupée par l'acteur
hide()	Cache l'acteur sans le supprimer. Après reset() il redevient visible

isInGrid()	Retourne True si l'acteur est situé à l'intérieur de la grille de jeu
isHorzMirror()	Retourne True si le sprite de l'acteur est retourné horizontalement (inversion gauche-droite)
isVertMirror()	Retourne True si le sprite de l'acteur est retourné verticalement (inversion haut-bas)
isMoveValid()	Retourne True si l'acteur reste dans la grille de jeu après un appel à move()
isNearBorder()	Retourne True si l'acteur se situe au bord de la grille de jeu
isVisible()	Returns True si l'acteur est visible
move()	Déplace l'acteur dans une cellule adjacente en conservant la direction actuelle
move(distance)	Idem, en spécifiant la distance parcourue par le déplacement
reset()	Méthode appelée lorsque l'acteur est ajouté à la GameGrid et lorsque le bouton « reset » est actionné
setCollisionCircle(spriteId, center, radius)	Définit une zone de collision circulaire de centre center et de rayon radius pour le sprite d'ID spriteId
setCollisionLine(spriteId, startPoint, endPoint)	Définit une zone de collision rectiligne définie par les points startPoint et endPoint pour le sprite d'ID spriteId
setCollisionRectangle(spriteId, center, width, height)	Définit une zone de collision rectangulaire de centre center, de largeur width et de hauteur height
setCollisionSpot(spriteId, spot)	Définit une zone de collision en un unique point de coordonnées spot pour le sprite d'ID spriteId
setCollisionImage(spriteId)	Définit les pixels non-transparents pour la collision. Seulement disponible si le partner utilise spot, line ou circle
setHorzMirror(True)	Effectue une inversion gauche/droite sur le sprite de l'acteur
setVertMirror(True)	Effectue une inversion haut/bas sur le sprite de l'acteur
setSlowDown(factor)	Ralentit d'un facteur factor les appels à la méthode act() des acteurs
setLocation(location)	Place l'acteur dans la cellule de coordonnées location au sein de la grille de jeu
setLocationOffset(point)	Décale le centre de l'image de sprite par rapport au centre de la cellule occupée par l'acteur. Ne change rien à la position de l'acteur dans la grille de jeu
setPixelLocation(location)	Place l'acteur aux coordonnées location exprimées en pixels (la position dans la grille et le décalage sont ajustés en conséquence)
setX(x)	Ajuste la coordonnée x à la valeur spécifiée
setY(y)	Ajuste la coordonnée y à la valeur spécifiée
show()	Rend visible le sprite d'ID 0
show(spriteId)	Rend visible le sprite d'ID spriteId
showNextSprite ()	Rend visible la prochaine image de sprite (spriteId est incrémenté de 1 modulo nbSprites). Ainsi, une fois que l'on a atteint le dernier sprite, on recommence au sprite d'id 0
showPreviousSprite()	Rend visible la précédente image de sprite (spriteId est décrémenté de 1 modulo nbSprites). Ainsi, une fois que l'on a atteint le premier sprite d'ID 0, en recommence avec le dernier d'ID nbSprite-1
removeSelf()	Supprime l'acteur. Il ne réapparaît plus lors d'un prochain reset()
reset()	Cette fonction est appelée par GameGrid.addActor() et lors d'un clic sur le bouton reset
turn(angle)	Change la direction du mouvement de l'acteur de l'angle indiqué en degrés, dans le sens des aiguilles de la montre

GGBackground

clear()	Efface l'arrière-fond actuel en remplaçant tous ses pixels par la couleur d'arrière-plan actuellement en usage
clear(color)	Idem, mais en remplaçant par la couleur d'arrière-plan color
drawCircle(center, radius)	Dessine un cercle de centre center et de rayon radius. Les coordonnées du centre ainsi que le rayon sont indiquées en pixels
drawLine(x1,y1, x2, y2)	Dessine un segment droit entre les points définis par les coordonnées pixel (x1, y1) et (x2, y2).
drawLine(pt1, pt2)	Idem, mais en donnant les coordonnées des points extrémaux sous forme d'objets Point (pixels)
drawPoint(pt)	Dessine un point de coordonnées indiquées par l'objet Point pt (pixels)
drawRectangle(pt1, pt2)	Dessine un rectangle de sommets supérieur gauche pt1 et inférieur droit pt2 (objets Point en pixels)
drawText(text, pt)	Écrit le texte text à la position indiquée par pt (objet Point en pixels)
fillCell(location, color)	Remplit la cellule de position location (coordonnées grille) avec la couleur color
fillCircle(center, radius)	Dessine un cercle plein de centre center et de rayon radius. Les coordonnées et le rayon sont donnés en pixels
getBgColor()	Retourne la couleur de fond actuelle
getColor(location)	Retourne la couleur de fond présente au centre de la cellule location. Les pixels de l'acteur ne sont pas pris en compte
save()	Sauvegarde la couleur de fond actuelle qui peut ensuite être restituée avec restore()
setBgColor(color)	Change la couleur d'arrière-fond
setFont(font)	Règle la police de caractères utilisée pour afficher les textes
setLineWidth(width)	Règle la largeur des lignes
setPaintColor(color)	Règle la couleur de dessin à color
setPaintMode()	Dessine sans tenir compte de la couleur de l'arrière-fond
setXORMode(color)	Passage en mode de dessin XOR. Deux appels successifs avec la même couleur color s'annulent
restore()	Restore l'arrière-fond précédemment sauvegardé avec save().

Location

Location(x, y)	Génère un objet location permettant une localisation à partir des coordonnées horizontales et verticales de la cellule au sein de la grille de jeux
Location(location)	Génère un objet location à partir de location (clone)
clone()	Retourne une nouvelle localisation avec les mêmes coordonnées grille
equals(location)	Retourne True si la localisation actuelle de l'acteur correspond à location
get4CompassDirectionTo(location)	Retourne une liste comportant les positions des quatre cellules adjacentes Ouest, Est, Nord, Sud par rapport à la position courante
getCompassDirectionTo(location)	Retourne une liste comportant les positions des huit cellules adjacentes, également en diagonale. Donc idem, mais en rajoutant encore Nord-Ouest, Nord-Est, Sud-Est et Sud-Ouest
getDirectionTo(location)	Retourne en degrés la direction à prendre pour aller de la position actuelle vers la position location donnée. 0 degrés = Est
getNeighbourLocation(direction)	Retourne la position d'une des huit cellules voisines. C'est la cellule la plus proche de la direction indiquée qui est retournée
getNeighbourLocations(distance)	Retourne une liste de toutes les cellules dont le centre se trouve à une distance inférieure ou égale à distance de la cellule actuelle
getX()	Retourne la position horizontale de la cellule occupée par l'acteur sur lequel getX() est invoquée
getY()	Retourne la position verticale de la cellule occupée par l'acteur sur lequel getY() est invoquée