

Pac-Man

Maze	<ul style="list-style-type: none"> • dimensions (width,height) • walls • pellets 	<ul style="list-style-type: none"> • a constructor which takes the dimensions as parameters and initializes those instance variable accordingly; the walls and pellets should be initialized as desired • getters getWidth, getHeight • reset, which creates and distributes a new set of pellets • createPacMan, which takes Pac-Man's size and direction of travel as parameters and returns a new PacMan object placed appropriately within the maze • createGhost, which takes the ghost's size, color, and direction of travel as parameters and return a new Ghost object placed appropriately within the maze • pelletsLeft, which returns whether or not there are any unconsumed pellets remaining • canMove, which takes Pac-Man as a parameter and returns true if Pac-Man is able to move forward (i.e. no walls in the way) and false otherwise (Pac-Man is blocked by a wall) • canMove, which takes a ghost as a parameter and returns true if the ghost is able to move forward (i.e. no walls in the way) and false otherwise (the ghost is blocked by a wall) • consumePellets, which takes Pac-Man as a parameter and consumes any pellets that Pac-Man is able to eat • draw, which takes a GraphicsContext as a parameter and draws the maze
------	---	---

- animation variables in PacMain are Pac-Man, (an array of) ghosts, and a maze

CSPC 124: Introduction to Programming • Spring 2024
98

Pac-Man

Maze	<ul style="list-style-type: none"> • dimensions (width,height) • walls • pellets 	<ul style="list-style-type: none"> • a constructor which takes the dimensions as parameters and initializes those instance variable accordingly; the walls and pellets should be initialized as desired • getters getWidth, getHeight • reset, which creates and distributes a new set of pellets • createPacMan, which takes Pac-Man's size and direction of travel as parameters and returns a new PacMan object placed appropriately within the maze • createGhost, which takes the ghost's size, color, and direction of travel as parameters and return a new Ghost object placed appropriately within the maze • pelletsLeft, which returns whether or not there are any unconsumed pellets remaining • canMove, which takes Pac-Man as a parameter and returns true if Pac-Man is able to move forward (i.e. no walls in the way) and false otherwise (Pac-Man is blocked by a wall) • canMove, which takes a ghost as a parameter and returns true if the ghost is able to move forward (i.e. no walls in the way) and false otherwise (the ghost is blocked by a wall) • consumePellets, which takes Pac-Man as a parameter and consumes any pellets that Pac-Man is able to eat • draw, which takes a GraphicsContext as a parameter and draws the maze
------	---	---

- start with creating PacMan and Ghost objects in PacMain
- eventually that should be done by createPacMan and createGhost, and those called instead of the constructors in PacMain

CSPC 124: Introduction to Programming • Spring 2024
99

Pac-Man

Maze	<ul style="list-style-type: none"> • dimensions (width,height) • walls • pellets 	<ul style="list-style-type: none"> parameters and return a new Ghost object placed appropriately within the maze • pelletsLeft, which returns whether or not there are any unconsumed pellets remaining • canMove, which takes Pac-Man as a parameter and returns true if Pac-Man is able to move forward (i.e. no walls in the way) and false otherwise (Pac-Man is blocked by a wall) • canMove, which takes a ghost as a parameter and returns true if the ghost is able to move forward (i.e. no walls in the way) and false otherwise (the ghost is blocked by a wall)
PacMan	<ul style="list-style-type: none"> • travel • power remaining (number of steps) • alive? 	<ul style="list-style-type: none"> • decreasePower, which decreases the power remaining by one • kill, which makes Pac-Man no longer alive • move, which moves Pac-Man one step in the current direction of travel • draw, which takes a GraphicsContext as a parameter and draws Pac-Man • isBlockedBy, which takes a wall as a parameter and returns true if Pac-Man is blocked from moving forward by the wall and false otherwise
Ghost	<ul style="list-style-type: none"> • color (type Color) • current direction of travel • alive? 	<ul style="list-style-type: none"> • kill, which makes the ghost no longer alive • move, which moves the ghost one step in the current direction of travel • draw, which takes a GraphicsContext as a parameter and draws the ghost • isBlockedBy, which takes a wall as a parameter and returns true if Pac-Man is blocked from moving forward by the wall and false otherwise

- it is legal to have two methods with the same name as long as the parameters are different (*overloading*)

CSPC 124: Introduction to Programming • Spring 2024
100

Pac-Man

Maze	<ul style="list-style-type: none"> • dimensions (width,height) • walls • pellets 	<ul style="list-style-type: none"> parameters and return a new Ghost object placed appropriately within the maze • pelletsLeft, which returns whether or not there are any unconsumed pellets remaining • canMove, which takes Pac-Man as a parameter and returns true if Pac-Man is able to move forward (i.e. no walls in the way) and false otherwise (Pac-Man is blocked by a wall) • canMove, which takes a ghost as a parameter and returns true if the ghost is able to move forward (i.e. no walls in the way) and false otherwise (the ghost is blocked by a wall)
PacMan	<ul style="list-style-type: none"> • travel • power remaining (number of steps) • alive? 	<ul style="list-style-type: none"> • decreasePower, which decreases the power remaining by one • kill, which makes Pac-Man no longer alive • move, which moves Pac-Man one step in the current direction of travel • draw, which takes a GraphicsContext as a parameter and draws Pac-Man • isBlockedBy, which takes a wall as a parameter and returns true if Pac-Man is blocked from moving forward by the wall and false otherwise
Ghost	<ul style="list-style-type: none"> • color (type Color) • current direction of travel • alive? 	<ul style="list-style-type: none"> • kill, which makes the ghost no longer alive • move, which moves the ghost one step in the current direction of travel • draw, which takes a GraphicsContext as a parameter and draws the ghost • isBlockedBy, which takes a wall as a parameter and returns true if Pac-Man is blocked from moving forward by the wall and false otherwise

- isBlockedBy (PacMan, Ghost) does the math of whether moving Pac-Man or the ghost would cause an overlap
- canMove (Maze) goes through each wall and asks Pac-Man/ghost whether they are blocked by that wall

CSPC 124: Introduction to Programming • Spring 2024
101

Pac-Man

Maze	<ul style="list-style-type: none"> • dimensions (width,height) • walls • pellets 	<ul style="list-style-type: none"> • parameters and return a new ghost object placed appropriately within the maze • pelletsLeft, which returns whether or not there are any unconsumed pellets remaining • canMove, which takes Pac-Man as a parameter and returns true if Pac-Man is able to move forward (i.e. no walls in the way) and false otherwise (Pac-Man is blocked by a wall) • canMove, which takes a ghost as a parameter and returns true if the ghost is able to move forward (i.e. no walls in the way) and false otherwise (the ghost is blocked by a wall) • consumePellets, which takes Pac-Man as a parameter and consumes any pellets that Pac-Man is able to eat
PacMan	<ul style="list-style-type: none"> • travel • power remaining (number of steps) • alive? 	<ul style="list-style-type: none"> • decreasePower, which decreases the power remaining by one • kill, which makes Pac-Man no longer alive • move, which moves Pac-Man one step in the current direction of travel • draw, which takes a GraphicsContext as a parameter and draws Pac-Man • isBlockedBy, which takes a wall as a parameter and returns true if Pac-Man is blocked from moving forward by the wall and false otherwise • canEat, which takes a pellet as a parameter and returns true if Pac-Man is in the right position to eat the pellet and false otherwise

- canEat (PacMan) does the math of whether Pac-Man can eat the pellet
- consumePellets (Maze) goes through each pellet, asks Pac-Man if the pellet can be eaten, and handles the pellet being eaten if so