**Pac-Man**

**Table of Content**

[**Introduction**](#_r1rv2wqw5qpx) **3**

[**Game Rules**](#_g2lrv9ja68wc) **4**

[Original Game](#_3yp4ehg4x9iz) 4

[My Modified Game](#_1eaklxas8mli) 5

[**Game Strategies**](#_e9859xrd8mjx) **5**

[**Source Code Referenced and Material Referenced**](#_8xxf4di340ud) **6**

[**Graphics**](#_55ed0yiuzd62) **6**

[**Code**](#_xdmfyirh2uhr) **7**

[**Data Structure**](#_77tyicx92g8g) **9**

[**Methods**](#_kjavtb8fv2x9) **9**

[First Method](#_vxnm02xmm5fl) 9

[Second Method](#_a69plm8un2cx) 10

[Third Method](#_i8uu503pity4) 10

[Fourth Method](#_ov3r99s4yrps) 10

[Fifth Method](#_lp1rop7m5zqn) 11

[Sixth Method](#_45ew9fpf4375) 12

[**Objects**](#_lja4k92l2sq5) **13**

[**Features**](#_ku7g4gpj8xqk) **14**

[**References**](#_h4nl15n1qkka) **15**

# **Introduction**

Pacman is a maze game and it comes in an arcade category which was developed in 1980 by Namco cooperation. The first name that was associated with this game is puck man and further, Puckman was changed with Pacman.

In this game, the player controls an eponymous character in a maze to collect all the pellets which reside in the maze by taking down their character to get passed through the pellets, and the character has to dodge the ghosts, if the ghost touches the character then the game is over. Each ghost has been moving with the help of computerized algorithms, also nowadays that we call artificial intelligence because they always choose the random path. In the original game, there are big dots called power pallets that turn the ghosts into the blue ghosts and in the meantime, Pacman’s character eats that blue ghost which can help Pacman to get some extra points and the ghost will move back to the original position. It was developed in Japan and the direction of the game is given by Toru Iwatani and developed with the help of 9 man teams. The main conception of the creation of the game is to make one game that can be appealing to both genders because most games are following the same theme which either is connected with war or sports. We have completed 20 years with the pacman game and it is quite good to have this kind of game. We can play this game when we are bored.

# Game Rules

## Original Game

1. In the original game, the eponymous character called Pacman to have to eat all pellets around the maze.
2. There are 4 power pallets that reside in each corner of the room and after eating that power pallet, four ghosts which are our enemies turn their color into blue, and the character called Pacman can eat the blue ghost, and that helps Pacman to gain some extra points.
3. The eponymous character has to dodge all the ghosts while the character is eating the normal pallets and if the ghost touches the character in meantime then the character loses their one life.
4. If the character reaches 10000 points while playing the game then it will gain one life.

## My Modified Game

## 

1. We also have one eponymous character that we have called Pacman and it also has to eat all pallets in the game and then the game will be over.
2. Our character also has to dodge the 4 ghosts in the maze by moving up-down, right-left.
3. The game will be over when the character will eat all the pallets without touching the ghosts in the game.

# Game Strategies

1. In the original game, we have to keep an eye on the closest one or red, & pink because they are faster than the other two. The character is faster than all the ghosts so the best way to stop you is to corner you or make you turn wrong. We have to see all the opponent activities like where they are and where they are moving.
2. Ensure that you are so smart, because ghosts always try to corner you so you have to take that path which has more paths and turn-offs.
3. You should use a power-up when at least 3 ghosts are in your area and concentrate on your points to grow up.
4. When a ghost regenerates after when you eat it in power-up so don’t come to the position where you are at the entrance of the gate where the ghost regenerates because you may lose one life.
5. Learn the paths and way where they are starting and try to understand the algorithm to come to a position where you will win each and every game.
6. Collect as many points as possible.

# Source Code Referenced and Material Referenced

1. Suarez, A. (2020). Pacman in Python Code. Its Source Code. <https://itsourcecode.com/free-projects/python-projects/pacman-in-python-code/>
2. Hattersley, L. (2018). Code PacMan in Python. Magpi. <https://magpi.raspberrypi.org/articles/code-pac-man-in-python>
3. Author, N. (2020). How to Win in Pac Man. Wikihow. <https://www.wikihow.com/Win-in-Pac-Man>
4. Author, n. (n.d.). Pacman Rules. Pacxon. <https://www.pacxon.net/pacman-rules.php>

# Graphics

A graphic is a depiction of an object in the form of a picture or a visual image. These are visual effects or patterns that are projected onto a surface, including a wall, canvas, monitor, journal, or object, in order to educate, demonstrate, or entertain.

In this game, we have used tiles, squares, spherical dots structures created with the help of a graphics creation library called turtle. After we import Turtle we can provide orders like forward, in reverse, right, left, and so on. These orders will draw various shapes. At the point when we consolidate search orders we can make numerous pleasant illustrations in the beneath model we will see some basic situations and afterward some Complex ones where decent designs are made.

We have also used the free games and random library to make it easy in making the vector graphics and computing random paths for running the ghost. We have made the ghosts, Pacman, dots, squares, pallets, etc.

# Code

**Main Code**

from turtle import \*

from random import choice

from freegames import floor, vector

if \_\_name\_\_ == '\_\_main\_\_':

# Starting State of score which is equal to 0

all\_states = {'score': 0}

# Making path not Visibility

pth\_turtle = Turtle(visible=False)

# Making turtle Writer not visible

writer\_turtle = Turtle(visible=False)

# Describe the Aim

aim\_machine = vector(5, 0)

# Making Pacman Character

pacman\_character = vector(-40, -80)

# Making 4 Ghosts

ghosts\_riders = [

[vector(-180, -160), vector(0, 5)],

[vector(100, -160), vector(-5, 0)],

[vector(-180, 160), vector(5, 0)],

[vector(100, 160), vector(0, -5)],

]

# These are the tiles which will make the game

game\_tiles = [

0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,

0, 1, 1, 1, 1, 1, 1, 1, 0, 1, 1, 1, 1, 1, 1, 1, 0, 0, 0, 0,

0, 1, 0, 0, 1, 0, 0, 1, 0, 1, 0, 0, 1, 0, 0, 1, 0, 0, 0, 0,

0, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 0, 0, 0, 0,

0, 1, 0, 0, 1, 0, 1, 0, 0, 0, 1, 0, 1, 0, 0, 1, 0, 0, 0, 0,

0, 1, 1, 1, 1, 0, 1, 1, 0, 1, 1, 0, 1, 1, 1, 1, 0, 0, 0, 0,

0, 1, 0, 0, 1, 0, 0, 1, 0, 1, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0,

0, 1, 0, 0, 1, 0, 1, 1, 1, 1, 1, 0, 1, 0, 0, 0, 0, 0, 0, 0,

0, 1, 1, 1, 1, 1, 1, 0, 0, 0, 1, 1, 1, 1, 1, 1, 0, 0, 0, 0,

0, 0, 0, 0, 1, 0, 1, 1, 1, 1, 1, 0, 1, 0, 0, 1, 0, 0, 0, 0,

0, 0, 0, 0, 1, 0, 1, 0, 0, 0, 1, 0, 1, 0, 0, 1, 0, 0, 0, 0,

0, 1, 1, 1, 1, 1, 1, 1, 0, 1, 1, 1, 1, 1, 1, 1, 0, 0, 0, 0,

0, 1, 0, 0, 1, 0, 0, 1, 0, 1, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0,

0, 1, 1, 0, 1, 1, 1, 1, 1, 1, 1, 1, 1, 0, 1, 1, 0, 0, 0, 0,

0, 0, 1, 0, 1, 0, 1, 0, 0, 0, 1, 0, 1, 0, 1, 0, 0, 0, 0, 0,

0, 1, 1, 1, 1, 0, 1, 1, 0, 1, 1, 0, 1, 1, 1, 1, 0, 0, 0, 0,

0, 1, 0, 0, 0, 0, 0, 1, 0, 1, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0,

0, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 0, 0, 0, 0,

0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,

0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,

]

# Setting the frame Size

setup(420, 420, 370, 0)

# Hiding Turtle

hideturtle()

# Making Turtle Tracer = False

tracer(False)

# Telling turtle to go to this position

writer\_turtle.goto(160, 160)

# Setting up the turtle color = White

writer\_turtle.color('white')

# This will update all the score on the window

writer\_turtle.write(all\_states['score'])

# This will listen all the key

listen()

# All keys funtions

onkey(lambda: do\_change(-5, 0), 'Left')

onkey(lambda: do\_change(0, -5), 'Down')

onkey(lambda: do\_change(5, 0), 'Right')

onkey(lambda: do\_change(0, 5), 'Up')

# Making Game World

game\_world()

# Move Elements

elements\_movement()

# Finish the Game

done()

# Data Structure

We have used the array-like data structure called list and more internal data structure of some libraries like turtle and PyGames to make and store some important information.

# Methods

## **First Method**

def making\_square(x, y):

path\_turtle.up()

path\_turtle.goto(x, y)

path\_turtle.down()

path\_turtle.begin\_fill()

for count in range(4):

path\_turtle.forward(20)

path\_turtle.left(90)

path\_turtle.end\_fill()

Description: This method will make squares into the board and we need x and y coordinates for making the square.

Parameters: x and y

Local Variables: None

Return: Nothing

## **Second Method**

def finding\_offset(game\_point):

x\_point = (floor(game\_point.x, 20) + 200) / 20

y\_point = (180 - floor(game\_point.y, 20)) / 20

game\_index = int(x\_point + y\_point \* 20)

return game\_index

Description: This method will find an offset of the game point.

Parameter: game point

Local Variables: x\_point, y\_point, game\_index

Return: game\_index

## Third Method

def check\_valid(game\_point):

game\_index = finding\_offset(game\_point)

if game\_tiles[game\_index] == 0:

return False

game\_index = finding\_offset(game\_point + 19)

if game\_tiles[game\_index] == 0:

return False

return game\_point.x % 20 == 0 or game\_point.y % 20 == 0

Description: This method helps in identifying the game point that it is valid is not.

Parameter: game\_point

Local Variables: game\_index

Return: True or False

## **Fourth Method**

def game\_world():

bgcolor('black')

path\_turtle.color('blue')

for game\_index in range(len(game\_tiles)):

game\_tile = game\_tiles[game\_index]

if game\_tile > 0:

x = (game\_index % 20) \* 20 - 200

y = 180 - (game\_index // 20) \* 20

making\_square(x, y)

if game\_tile == 1:

path\_turtle.up()

path\_turtle.goto(x + 10, y + 10)

path\_turtle.dot(2, 'white')

Description: This method will make the game world, By making squares and objects into the virtual board.

Parameter: None

Local Variables: game\_tile, x, and y

Return: Nothing

## **Fifth Method**

def elements\_movement():

"Move pacman and all ghosts."

writer\_turtle.undo()

writer\_turtle.write(all\_states['score'])

clear()

if check\_valid(pacman\_character + aim\_machine):

pacman\_character.move(aim\_machine)

game\_index = finding\_offset(pacman\_character)

if game\_tiles[game\_index] == 1:

game\_tiles[game\_index] = 2

all\_states['score'] += 1

x = (game\_index % 20) \* 20 - 200

y = 180 - (game\_index // 20) \* 20

making\_square(x, y)

up()

goto(pacman\_character.x + 10, pacman\_character.y + 10)

dot(20, 'yellow')

for pnt, crse in ghosts\_riders:

if check\_valid(pnt + crse):

pnt.move(crse)

else:

options = [

vector(-5, 0),

vector(0, -5),

vector(5, 0),

vector(0, 5),

]

plan = choice(options)

crse.x = plan.x

crse.y = plan.y

up()

goto(pnt.x + 10, pnt.y + 10)

dot(20, 'red')

update()

for pnt, crse in ghosts\_riders:

if abs(pacman\_character - pnt) < 20:

return

ontimer(elements\_movement, 100)

Description: This method helps in the movement of the Pacman character with the help of keys and generates an algorithm for the movement of all ghosts.

Parameter: None

Local Variables: game\_index

Return: Nothing

## Sixth Method

def do\_change(x, y):

if check\_valid(pacman\_character+ vector(x, y)):

aim\_machine.x = x

aim\_machine.y = y

Description: This method will help in moving the aim of the Pacman character if the aim is valid.

Parameter: x, and y

Local Variables: None

Return: Nothing

# Objects

We have not made any kind of explicit classes in the game so in that, we didn’t have any kind of explicit objects in the program.

All we have is the object of the libraries like Turtle, Random, and Freegames.

We have imported all function objects of turtle in the code and use some objects like a writer, hide turtle, listen, on key, etc.

We have imported a choice function object from a random library to choose the randomness in the program for running the ghost in the random path.

We also have imported floor function objects and vector objects from the free games library.

This all object helps in the development of the game.

Install:

* Pip install freegames - for downloading the freegames library
* Pip install turtle - for downloading the turtle library
* Pip install random - for downloading the random library

# Features

Sound and Sound Effects: In that time of video games, the sound is crucial to the overall performance. The ultimate gaming atmosphere is greatly influenced by sound effects. This includes creating entire libraries of custom effects of sound to provide the game a sense of realism and uniqueness. The origin of the sound is mimicking the phrase ‘paku paku taberu’ in that paku paku means making a sound of snapping mouth and taberu means ‘to eat’. All the game themes are associated with girls and women. An original Pacman sound made by someone 13 years before the game was developed like it is a glory of the arcade games sound, it developed in 1967 in a studio of France called Di Fonologia Musicale Di Firenze experiment with sound with technologies like they released GE-115 Computer Concerto computer which is used to generate the excerpts from Bach and Paganini. A sound named mixed Paganini is kind of similar to the intro of the Pacman and the sound of the bleep and bloop that comes out when the pacman character goes for hunting.

Sound effects are used for a variety of uses, including:

* Creating an atmosphere
* Realistic touches
* Giving hints about the environment
* Increasing the fun factor
* Create interactive and screen responses
* Create a distinctive style

And the Pacman game is all associated with these sound themes, that is kind of indulging with the sound.

FILE I/O: A file is a term for a specific location on a computer's non-volatile storage medium, such as a hard drive. For reading and editing scripts, the I/O library is used. This is the best feature where the scores and initials of the players save and load. The size of this game is very low like in kb’s but in the ’80s KB’s storage is like a very big amount. It can be directly stored in the magnetic tape storage or directly stored in a floppy disk and can be played with by inserting a floppy disk in the PC. nowadays it is like a joke to store the game, google gives in free to play the game in play game platform. We can install something about one lakh Pacman kind of game in one single mobile. The score system is also stored in in-game memory.

SCORE SYSTEM: A score is an arbitrary quantity associated with an individual or team in sports. The score is normally expressed in the abstract unit of points (with the exception of game shows, where scores are often expressed in currency units), and activities in the game will increase or lower the score of various players. Most games with a score use it as a subjective measure of game success, and competitive games use it as well. Pacman score system is good like we have to collect pallets and dodge the ghosts to get some point and eat the power pallets and eat the ghost to get the extra 200, 400, 800, & 1600 points and when we raise the 10000 points while continuing the game then you can get one extra life. Complexity is like searching through generating hashing will take linear time but in that comparison is difficult so we will ensure that complexity will not go above n^2.

# References

* Sokol, Z. (n.d). Someone Made The Pac-Man Soundtrack 13 Years Before The Game Was Created. Vice. <https://www.vice.com/en/article/qkwpnm/someone-made-an-arcade-game-soundtrack-13-years-before-pac-man>
* Suarez, A. (2020). Pacman in Python Code. It Source Code. <https://itsourcecode.com/free-projects/python-projects/pacman-in-python-code/>
* Hattersley, L. (2018). Code Pac-Man in Python. Magpi. <https://magpi.raspberrypi.org/articles/code-pac-man-in-python>
* Author, N. (2020). How to Win in Pac Man. Wikihow. <https://www.wikihow.com/Win-in-Pac-Man>
* Author, N. (n.d.). Pacman Rules. Pacxon. <https://www.pacxon.net/pacman-rules.php>
* Prisco, J. (n.d.). Pac-Man at 40: The eating icon that changed gaming history. CNN. <https://edition.cnn.com/style/article/pac-man-40-anniversary-history/index.html#:~:text=But%20the%20creator%20of%20%22Pac,to%20try%20something%20completely%20different.&text=%22When%20I%20started%20drafting%20up,firm%20Namco%20at%20the%20time>.