

SNAKE GAME

A Course Based Project Submitted in Partial Fulfilment of the Requirement for the
Award of the degree of

BACHELOR OF TECHNOLOGY

COMPUTER SCIENCE AND ENGINEERING

Submitted by

DANDIGA SHRAVYA



DEPARTMENT OF CSE-CYS, DS & (AI &DS)

VALLURUPALLI NAGESWARARAO VIGNANA JYOTHI

INSTITUTE OF ENGINEERING AND TECHNOLOGY

(An Autonomous Institute, NAAC Accredited With 'A++' Grade, NBA
Accredited, Approved by AICTE, New Delhi, Affiliated to JNTUH)

VALLURUPALLI NAGESWARARAO VIGNANA JYOTHI
INSTITUTE OF ENGINEERING AND TECHNOLOGY

(An Autonomous Institute)



CERTIFICATE

This is to Certify that DANDIGA SHRAVYA (22075A6202) has successfully completed their project work at CSE CYS, DS & (AI & DS) Department of VNRVJIET, Hyderabad entitled “SNAKE GAME” in partial fulfilment of the requirements for the award of the Bachelor of Technology degree during the Academic year 2022-2023

Project Guide

Mrs.E. Lalitha
Assistant Prof. and Internal Guide
Dept. of CSE-CYS, DS and AI&DS
VNRVJIET

Head of Department

Dr.Mr. Rajasekhar
Prof. and Head
Dept. of CSE-CYS, DS and AI&DS
VNRVJIET

DECLARATION

This is to certify that the project work entitled "SNAKE GAME "submitted in VNR Vignana Jyothi Institute of Engineering & Technology in partial fulfilment of requirement for the award of Bachelor of Technology in Computer Science and Engineering. It is a Bonafide report of the work carried out by us under the guidance and supervision of Mrs.E.Lalitha (Assistant Professor), Department of CSE-CYS,DS,AI&DS, VNRVJIET. To the best of our knowledge, this report has not been submitted in any form to any university or institution for the award of any degree or diploma.

DANDIGA SHRAVYA
22075A6202
CSE-CYBERSECURITY

ACKNOWLEDGEMENT

Behind every achievement lies the heartfelt gratitude to those who activated in completing the project. To them we lay the words of gratitude within us.

We are indebted to our venerable principal Dr. C.D. NAIDU for this inflicting devotion, which led us to complete this project. The support, encouragement given by him and his motivation led us to complete the project.

We express our sincere thanks to internal guide Mrs.E.Lalitha and also Head of the Department Dr. M. RAJA SHEKHAR for having provided us a lot of facilities to undertake the project work and guide us to complete the project.

We take the opportunity to express thanks to our faculty of the Dept. of COMPUTER SCIENCE AND ENGINEERING-CYBER SECURITY and remaining members of our college VNR VIGNANA JYOTHI INSTITUTE OF ENGINEERING AND TECHNOLOGY who extended their valuable support in helping us to complete the project in time.

DANDIGA SHRAVYA 22075A6202

ABSTRACT

The Project entitled 'Snake Game' is a simple console application without graphics. In this project, you can play the popular "Snake Game" just like you played it elsewhere. You have to use the up, down, right or left arrows to move the snake. Foods are provided at the several co-ordinates of the screen for the snake to eat. Every time the snake eats the food, its length will be increased by one element along with the score. I have used Python as Programming language for writing the code for the project and Jupyter Notebook for writing the programs. Operating system used Windows 10

CONTENTS

Topics	Page No.
1. ACKNOWLEDGEMENT	4
2. ABSTRACT	5
3. INTRODUCTION	7
4. LIBRARIES	8- 11
5. CODE	12 - 17
6. OUTPUT	18
7. CONCLUSION	19

INTRODUCTION

Playing games is fun and exciting it gives us relief from stress and unwind from our stressful works. Many of us spend our vacant time or others that use most of their time in playing and exploring new games. Today, with the rapid development of technology we have, games that are rising up together with it. Nowadays with the technology we have many games that are developing for computer most specifically for windows. With the high technology equipped with these computers, computer games become robust and attract many people to buy or have this gadget for them to experience what's inside it which makes it a trend for the new generation of gadget. Snake game is a computer action game; whose goal is to control a snake to move and collect food in a map. It has been around since the earliest days of home computing and has re-emerged in recent years on mobile phones. It isn't the world's greatest game, but it does give you an idea of what you can achieve with a simple python program, and perhaps the basis by which to extend the principles and create more interesting games on your own. To move the snake, use 'up arrow' for up, 'down arrow' for down, 'left arrow' for left and 'right arrow' for right. Press 'Q' to exit the game at any time, press 'C' to continue the game. The aim of the game is to collect the dots (food) and avoid the obstacles (crosses borders). As you collect the food, the snake gets longer. The score also increases. There is no concept of lives. Once you hit an obstacle, that's it, game over

LIBRARIES

Pygame

- Py game is a cross-platform set of python modules designed for writing video games. It includes computer graphics and sound libraries designed to be used with the Python Programming language.

- To install the library, you can use pip installer from the command line:

```
pip install pygame import pygame
```

Python time module

- Python has a module named time to handle time-related tasks. To use functions defined in the module, we need to import the module first. Here's how:

```
import time
```

Python random module

- Python has a built-in module that you can use to make random numbers.

```
import random
```

OVERVIEW TECHNOLOGY:

PYTHON

Python is an object-oriented, high level language, interpreted, dynamic and multipurpose programming language. Python is easy to learn yet powerful and versatile scripting language which makes it attractive for Application Development. Python's syntax and dynamic typing with its interpreted nature, make it an ideal language for scripting and rapid application development in many areas. Python supports multiple programming pattern, including object oriented programming, imperative and functional programming or procedural styles. Python is not intended to work on special area such as web

programming. That is why it is known as multipurpose because it can be used with web, enterprise, 3D CAD etc. We don't need to use data types to declare variable because it is dynamically typed so we can write `a=10` to declare an integer value in a variable. Python makes the development and debugging fast because there is no compilation step included in python development and edit-test-debug cycle is very fast.

PY-GAME

Python is the most popular programming language or nothing wrong to say that it is the next- generation programming language. In every emerging field in computer science, Python makes its presence actively. Python has vast libraries for various fields such as Machine Learning (Numpy,Pandas, Matplotlib), Artificial intelligence (Pytorch, TensorFlow), and Game development (Pygame,Pyglet).Game programming is very rewarding nowadays and it can also be used in advertising and as a teaching tool too. Game development includes mathematics, logic, physics, AI, and much more and it can be amazingly fun. In python, game programming is done in pygame and it is one of the best modules for doing so.

Installing pygame:

The first thing you will need to do in order to create games using Pygame is to install it on your systems. To do that, you can simply use the following command: `pip install pygame`

FUNCTION DESCRIPTION

Once that is done, just import Pygame and start our game development. Before moving on, take a look at the Pygame functions that have been used in this Snake Game along with their descriptions.

Function Description

- `init()` : Initializes all of the imported Pygame modules (returns a tuple indicating

success and failure of initializations)

- `display.set_mode()` : Takes a tuple or a list as its parameter to create a surface(tuple preferred)
- `update()` : Updates the screen
- `quit()` : Used to uninitialize everything
- `setcaption()` : Will set the caption text on the top of the display screen
- `event.get()` : Returns list of all events
- `Surface.fill()` : Will fill the surface with a solid color
- `ime.Clock()` : Helps track time
- `font.SysFont()` : Will create a Pygame font from the System font resources

IMPLEMENTATION STEPS

1. Installing pygame
2. Create the screen
3. Create the Snake
4. Moving the Snake
5. Game over when snake hits the boundaries
6. Adding the Food
7. Increasing the length of the snake
8. Displaying the score

WORKING ALGORITHM

Let's look at how a program to run the whole game might look:

1. Draw the playing area with bounding rectangle, set the counter to zero and display it.

2. Draw the snake in a starting position.
3. Draw the food in a starting location.
4. On user input, change snake direction.
5. Move the snake one move
6. If the snake is over food, eat it, increase the score, grow, move the food,
7. else if the snake is over in a wall, die.
8. Go back to 4.
9. Until the snake die

CODE FOR SNAKE GAME

```
import pygame
import time
import random

snake_speed = 15

# Window size
window_x = 720
window_y = 480

# defining colors
black = pygame.Color(0, 0, 0)
white = pygame.Color(255, 255, 255)
red = pygame.Color(255, 0, 0)
green = pygame.Color(0, 255, 0)
blue = pygame.Color(0, 0, 255)

# Initialising pygame
pygame.init()

# Initialise game window
pygame.display.set_caption('GeeksforGeeks Snakes')
game_window = pygame.display.set_mode((window_x, window_y))

# FPS (frames per second)
controller_fps = pygame.time.Clock()

# defining snake default position
snake_position = [100, 50]

# defining first 4 blocks of snake
body snake_body = [[100, 50],
                    [90, 50],
                    [80, 50],
                    [70, 50]
                    ]

# fruit position
fruit_position = [random.randrange(1, (window_x//10)) * 10,
```

```

        random.randrange(1, (window_y//10)) * 10]

fruit_spawn = True

# setting default snake direction
towards # right
direction =
'RIGHT' change_to
= direction

# initial score

score = 0

# displaying Score function
def show_score(choice, color, font, size):

    # creating font object score_font
    score_font = pygame.font.SysFont(font, size)

    # create the display surface
    object # score_surface
    score_surface = score_font.render('Score : ' + str(score), True, color)

    # create a rectangular object for
    the text # surface object
    score_rect = score_surface.get_rect()

    # displaying text
    game_window.blit(score_surface,
    score_rect)

# game over
function def
game_over():

```

```

# creating font object my_font
my_font = pygame.font.SysFont('times new roman', 50)

# creating a text surface on
which text # will be drawn
game_over_surface =
    my_font.render( 'Your Score is : '
        + str(score), True, red)

# create a rectangular object for
the text # surface object
game_over_rect = game_over_surface.get_rect()

# setting position of the text
game_over_rect.midtop = (window_x/2, window_y/4)

# blit will draw the text on screen
game_window.blit(game_over_surface, game_over_rect)
pygame.display.flip()

# after 2 seconds we will quit the program
time.sleep(2)

# deactivating pygame
library pygame.quit()

# quit the
program quit()

```

```

# Main
Function while
True:

    # handling key events
    for event in pygame.event.get():
        if event.type ==
            pygame.KEYDOWN:
                if
                    event.key == pygame.K_UP:
                        change_to = 'UP'
                if
                    event.key ==
                        pygame.K_DOWN:
                            change_to
                                = 'DOWN'
                if
                    event.key ==
                        pygame.K_LEFT:
                            change_to = 'LEFT'
                if
                    event.key ==
                        pygame.K_RIGHT:
                            change_to
                                = 'RIGHT'

    # If two keys pressed simultaneously
    # we don't want snake to move
    into two # directions
    simultaneously
    if change_to == 'UP' and direction !=
        'DOWN': direction = 'UP'
    if change_to == 'DOWN' and direction !=
        'UP': direction = 'DOWN'
    if change_to == 'LEFT' and direction !=
        'RIGHT': direction = 'LEFT'
    if change_to == 'RIGHT' and direction !=
        'LEFT': direction = 'RIGHT'

    # Moving the
    snake if direction
    == 'UP':
        snake_position[1] -=
    10 if direction ==
    'DOWN':
        snake_position[1] +=
    10 if direction ==
    'LEFT':
        snake_position[0] -=

```

```

10 if direction ==
'RIGHT':
    snake_position[0] += 10

# Snake body growing mechanism
# if fruits and snakes collide then
scores # will be incremented by
10
snake_body.insert(0, list(snake_position))
if snake_position[0] == fruit_position[0] and snake_position[1] ==
    fruit_position[1]: score += 10
    fruit_spawn = False
else:
    snake_body.pop()

if not fruit_spawn:
    fruit_position = [random.randrange(1, (window_x//10)) * 10,
        random.randrange(1, (window_y//10)) * 10]

```



```

fruit_spawn = True
game_window.fill(black)

for pos in snake_body:
    pygame.draw.rect(game_window, green,
                     pygame.Rect(pos[0], pos[1], 10, 10))
pygame.draw.rect(game_window, white, pygame.Rect(
    fruit_position[0], fruit_position[1], 10, 10))

# Game Over conditions
if snake_position[0] < 0 or snake_position[0] >
    window_x-10: game_over()
if snake_position[1] < 0 or snake_position[1] >
    window_y-10: game_over()

# Touching the
snake body for
block in
snake_body[1:]:
    if snake_position[0] == block[0] and snake_position[1] ==
        block[1]: game_over()

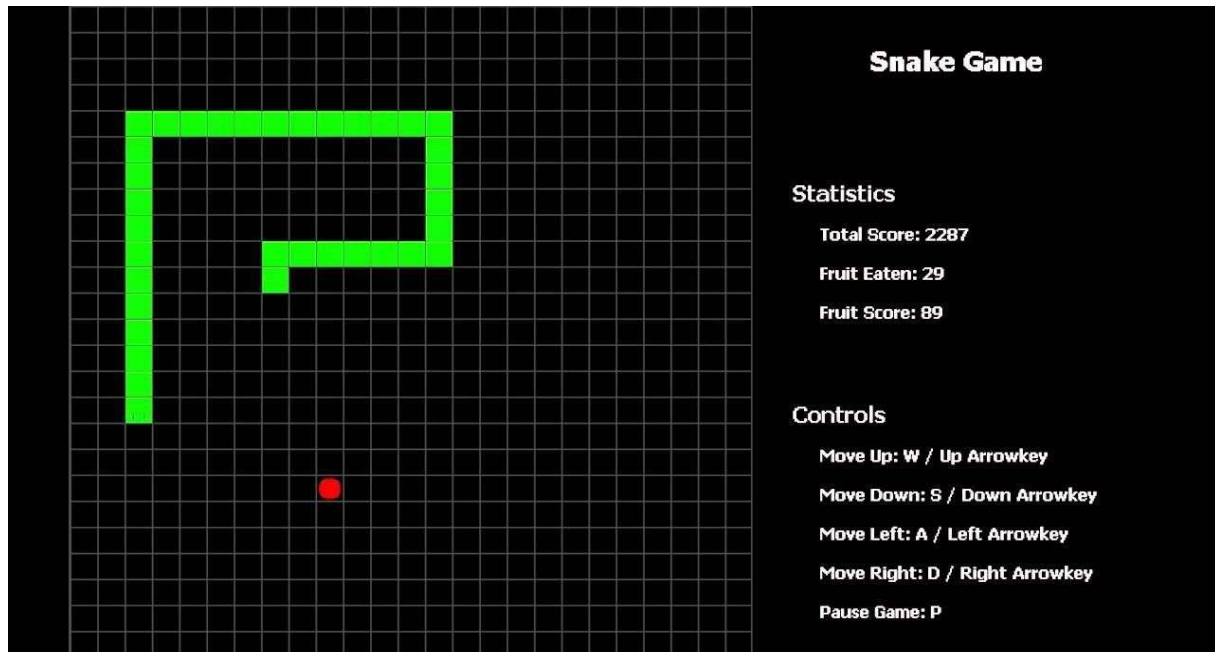
# displaying score countinuously
show_score(1, white, 'times new
roman', 20)

# Refresh game
screen
pygame.display.update()

# Frame Per Second /Refresh Rate
fps.tick(snake_speed)

```

OUTPUT



CONCLUSION

It isn't the world's greatest game, but it does give you an idea of what you can achieve with a relatively simple python programming, and perhaps the basis by which to extend the principles and create more interesting games on your own.