15/10/25 Task-12 Simulate Gaming Concepts using Python

AIM :- To Simulate Gaming concepts using pygame.

Snake Game :-

Problem 1 :- Write a python program to create a Snake Game using pygame package.

Conditions :-

(i) Set the window size

(ii) Create a Snake

(iii) Make the snake to move in the directions when left, right, down and up key is pressed.

(iv) When the snake hits the fruit, increase the score by 10.

(v) If the snake hits the window. Game over

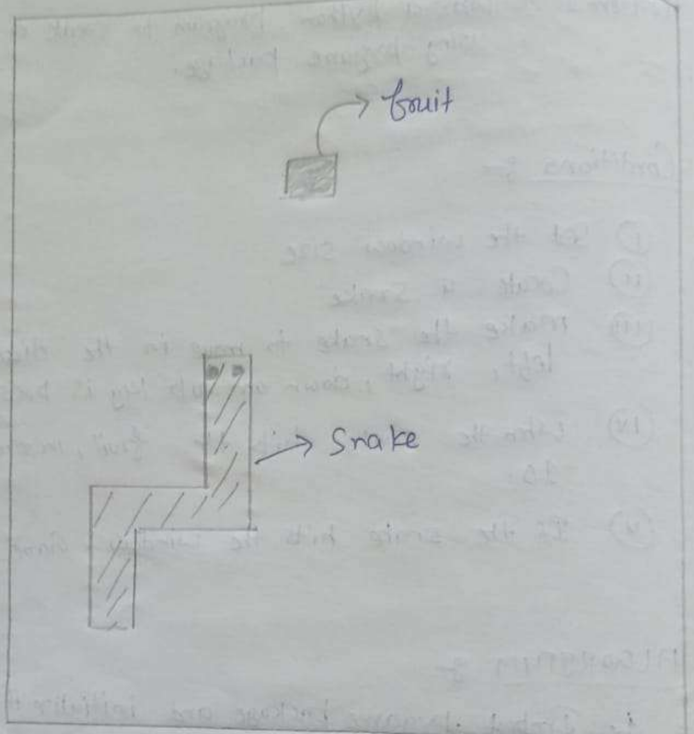
ALGORITHM :-

1. Import pygame package and initialize it.
2. Define the window size and title.
3. Create a Snake class which initializes the snake position, color and movement.
4. Create a functions to check if the snake collides with the fruit and increase the score.
5. Create a function to update the gamedisplay and draw the snake and fruit.
6. Create a game loop to continue
7. End the Game.

Program :-

```
# importing libraries
 import pygame
 import time
 Snake _ speed = 15
window_X = 720
window_Y = 480

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 )
Pygame . init()

Pygame. display set

fps = pygame. time. Clock ( )
 snake_position = [[100, 50], [90, 50] , [80, 50], [70,50],]
fruit_ position = [ random. randrange ( 1, window_X//10]] *b
           random. randrange ( 1, (window_Y //10.)) * 10]

 fruit_spawn = True

direction = 'RIGHT'
Charge _ to = direction

 score = 0
def show _score (choice, color , font, size):
 Score _ font = Pygame . font. SysFont ( Font, Size )
 Score _ surface = score. font. render
```

Output :



```
my-font = Pygame- font-SysFont ("times new roman", 50)
  Your Score is : ' + str(score). True, red)
  game-over- rect = game-over- surface - get- rect ()
  game-over- rect.midtop = (window-X/2, window - Y/4):

        for event in Pygame. event. get():
            if event.type == pygame-'KEYDOWM:
                if event. key = = Pygame. K-UP:
                    Change-to = 'DOWN'
                if event- Key = = Pygame-K-LEFT:
                    Change-to = 'LEFT'
                if event- key = = Pygame- K- RIGHT:
                    Change-to = 'RIGHT'

         if direction =='UP':
                snake- position [1]=10
          if direction = = 'DOWN':
                snake- position [1] += 10
            if direction = = 'LEFT':
                snake- position [1]- = 10
            if direction = = 'RIGHT':
              snake - position [0] += 10

        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))
```

```
if Snake Position [0]<0 :
            game_over [)

if Snake_ position [1)<0
        game_ over [].

Show score (1, white' times new roman, 20)

Pygame . display. update 1)

fps. tick ( snake - speed ,
```

Completed

TECH -	
EX NO.	12
PERFORMANCE (5)	5
RESULT AND ANALYSIS (3)	3
VIVA VOCE (3)	3
RECORD (4)	4
TOTAL (15)	15
SIGN WITH DATE	

RESULT :- Therefore, Stimulation of Gaming concept using Python is Comple -ted.