```
import pygame
import random

pygame.display.set_caption("Jogo da Vida")
pygame.init()

BLACK = (0, 0, 0)
GREY = (128, 128, 128)
YELLOW = (255, 255, 0)

WIDTH, HEIGHT = 800, 800
TILE_SIZE = 20
GRID_WIDTH = WIDTH // TILE_SIZE
GRID_HEIGHT = HEIGHT // TILE_SIZE
FPS = 60

screen = pygame.display.set_mode((WIDTH, HEIGHT))

clock = pygame.time.Clock()

def gen(num):
    return set([(random.randrange(0, GRID_HEIGHT), random.randrange(0,
GRID_WIDTH)) for _ in range(num)])

def draw_grid(positions):
    for position in positions:
        col, row = position
        top_left = (col * TILE_SIZE, row * TILE_SIZE)
        pygame.draw.rect(screen, YELLOW, (*top_left, TILE_SIZE,
TILE_SIZE))

    for row in range(GRID_HEIGHT):
        pygame.draw.line(screen, BLACK, (0, row * TILE_SIZE), (WIDTH, row
* TILE_SIZE))

    for col in range(GRID_WIDTH):
        pygame.draw.line(screen, BLACK, (col * TILE_SIZE, 0), (col *
TILE_SIZE, HEIGHT))

def adjust_grid(positions):
    all_neighbors = set()
    new_positions = set()

    for position in positions:
        neighbors = get_neighbors(position)
        all_neighbors.update(neighbors)

        neighbors = list(filter(lambda x: x in positions, neighbors))

        if len(neighbors) in [2, 3]:
            new_positions.add(position)

    for position in all_neighbors:
        neighbors = get_neighbors(position)
        neighbors = list(filter(lambda x: x in positions, neighbors))

        if len(neighbors) == 3:
            new_positions.add(position)
```

```
    return new_positions

def get_neighbors(pos):
    x, y = pos
    neighbors = []
    for dx in [-1, 0, 1]:
        if x + dx < 0 or x + dx > GRID_WIDTH:
            continue
        for dy in [-1, 0, 1]:
            if y + dy < 0 or y + dy > GRID_HEIGHT:
                continue
            if dx == 0 and dy == 0:
                continue

            neighbors.append((x + dx, y + dy))

    return neighbors

def main():
    running = True
    playing = False
    count = 0
    update_freq = 120

    positions = set()
    while running:
        clock.tick(FPS)

        if playing:
            count += 1

        if count >= update_freq:
            count = 0
            positions = adjust_grid(positions)

        pygame.display.set_caption("Jogando" if playing else "Pausado")

        for event in pygame.event.get():
            if event.type == pygame.QUIT:
                running = False

            if event.type == pygame.MOUSEBUTTONDOWN:
                x, y = pygame.mouse.get_pos()
                col = x // TILE_SIZE
                row = y // TILE_SIZE
                pos = (col, row)

                if pos in positions:
                    positions.remove(pos)
                else:
                    positions.add(pos)

            if event.type == pygame.KEYDOWN:
                if event.key == pygame.K_SPACE:
                    playing = not playing

                if event.key == pygame.K_c:
```

```
                    positions = set()
                    playing = False
                    count = 0

                if event.key == pygame.K_g:
                    positions = gen(random.randrange(4, 10) * GRID_WIDTH)

        screen.fill(GREY)
        draw_grid(positions)
        pygame.display.update()

    pygame.quit()

if __name__ == "__main__":
    main()
```