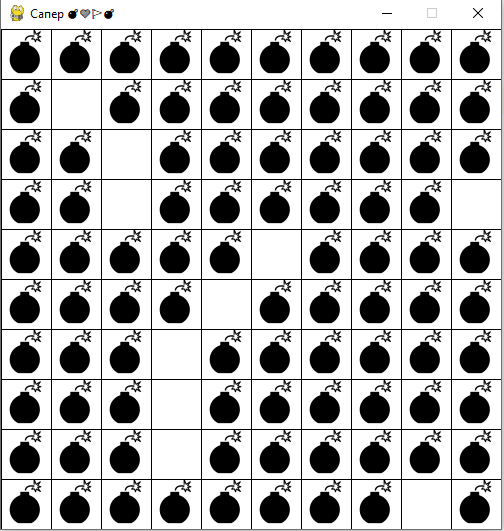
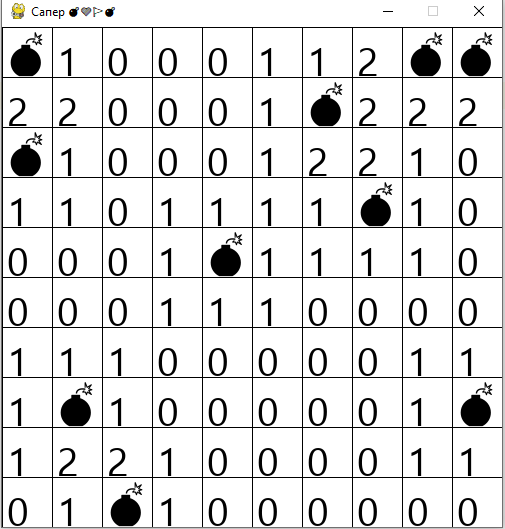
Сапер

import time  
  
from pygame import display, init,

draw, event, MOUSEBUTTONDOWN, font, QUIT  
from random import randrange  
  
init()  
font.init()  
  
SIZE = (500, 500)  
  
SCREEN = display.set\_mode(SIZE)  
display.set\_caption("Сапер 💣🧡🚩💣")  
*# for x in (200, 400):  
# draw.line(SCREEN, 'white', (x, 0), (x, 600), 5)  
# draw.line(SCREEN, 'white', (0, x), (600, x), 5)*for x in range(1, 500, 50):  
 for y in range(1, 500, 50):  
 draw.rect(SCREEN, 'white', (x, y, 49, 49))  
 draw.rect(SCREEN, 'white', (x, y, 49, 49))  
display.update()  
*# print('◕ ♥ ⚫ 🧡🚩 ❓💣')  
# print('/U1F6A9')*font = font.SysFont('segoe-ui-symbol', 40)  
  
  
def draw\_field():  
 n = 0  
 lst\_mines = {}  
 while n < 10:  
 x,y = randrange(5, 500, 50), randrange(0, 500, 50)  
 print(x,y)  
 char\_render = font.render('💣', True, 'black')  
 SCREEN.blit(char\_render, (x, y))  
 n+=1  
 display.update()  
  
  
draw\_field()  
while True:  
 for e in event.get():  
 if e.type == QUIT:  
 exit()  
 if e.type == MOUSEBUTTONDOWN:  
 x, y = e.pos  
 *# user\_make\_step(x, y)*

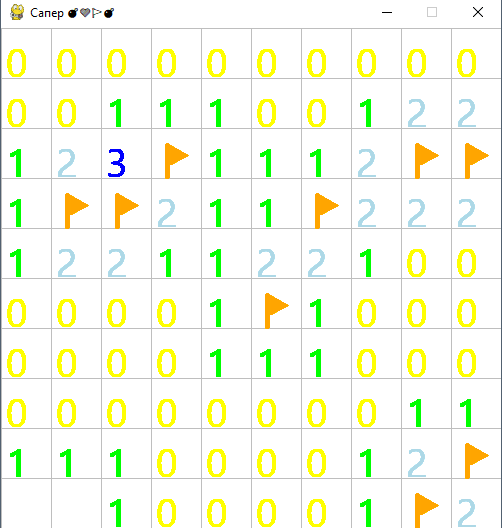
def draw\_field():  
 n = 0  
 lst\_mines = {}  
 while n < 90:  
 coords = randrange(5, 500, 50), randrange(0, 500, 50),  
 if coords not in lst\_mines:  
 lst\_mines[coords] = True  
 char\_render = font.render('💣', True, 'black')  
 SCREEN.blit(char\_render, coords)  
 n+=1  
 display.update()

Программируем счет бомб

from pygame import display, init, draw, event,

MOUSEBUTTONDOWN, font, QUIT  
from random import randrange  
  
font.init()  
  
SIZE = (500, 500)  
SCREEN = display.set\_mode(SIZE)  
display.set\_caption("Сапер 💣🧡🚩💣")  
lst\_mines = {}  
font = font.SysFont('segoe-ui-symbol', 40)  
  
for x in range(1, 500, 50):  
 for y in range(1, 500, 50):  
 draw.rect(SCREEN, 'white', (x, y, 49, 49))  
 draw.rect(SCREEN, 'white', (x, y, 49, 49))  
display.update()  
  
  
def draw\_field():  
 n = 0  
 while n < 10:  
 coords = randrange(5, 500, 50), randrange(5, 500, 50),  
 if coords not in lst\_mines:  
 lst\_mines[coords] = True  
 char\_render = font.render('💣', True, 'black')  
 SCREEN.blit(char\_render, coords)  
 n += 1  
 display.update()  
  
 for x in range(5, 500, 50):  
 for y in range(5, 500, 50):  
 if (x, y) in lst\_mines:  
 continue  
 char = str(count\_mines(x, y))  
 char\_render = font.render(char, True, 'black')  
 SCREEN.blit(char\_render, (x, y))  
 n += 1  
 display.update()  
 count\_mines(x, y)  
  
  
def count\_mines(x, y):  
 c = 0  
 for i in range(x - 50, x + 51, 50):  
 for j in range(y - 50, y + 51, 50):  
 c += lst\_mines.get((i, j), 0)  
 return c  
  
  
draw\_field()  
while True:  
 for e in event.get():  
 if e.type == QUIT:  
 exit()  
 if e.type == MOUSEBUTTONDOWN:  
 x, y = e.pos  
 *# user\_make\_step(x, y)*

Весь листинг:

import time  
  
from pygame import display, init, draw, event, MOUSEBUTTONDOWN, font, QUIT, mouse  
from random import randrange  
  
font.init()  
  
SIZE = (500, 500)  
SCREEN = display.set\_mode(SIZE)  
GAME = True  
display.set\_caption("Сапер 💣🧡🚩💣")  
font = font.SysFont('segoe-ui-symbol', 40)  
SCREEN.fill('grey')  
  
lst\_mines = {}  
control\_field\_dict = {}  
  
for x in range(1, 500, 50):  
 for y in range(1, 500, 50):  
 draw.rect(SCREEN, 'white', (x, y, 49, 49))  
 draw.rect(SCREEN, 'white', (x, y, 49, 49))  
display.update()  
  
  
def draw\_cell(x, y, char, color):  
 char\_render = font.render(str(char), True, color)  
 SCREEN.blit(char\_render, (x, y))  
 display.update()  
 if char == '💣' and color == 'red':  
 time.sleep(2)  
 SCREEN.fill('red')  
 display.update()  
  
  
def draw\_field():  
 n = 0  
 while n < 10:  
 coords = randrange(5, 500, 50), randrange(5, 500, 50),  
 if coords not in lst\_mines:  
 lst\_mines[coords] = True  
 draw\_cell(\*coords, '💣', 'white')  
 n += 1  
  
 for x in range(5, 500, 50):  
 for y in range(5, 500, 50):  
 if (x, y) in lst\_mines:  
 control\_field\_dict[(x, y)] = '💣'  
 continue  
 char = count\_mines(x, y)  
 control\_field\_dict[(x, y)] = char  
 draw\_cell(x, y, char, 'white')  
  
 count\_mines(x, y)  
  
  
def count\_mines(x, y):  
 c = 0  
 for i in range(x - 50, x + 51, 50):  
 for j in range(y - 50, y + 51, 50):  
 c += lst\_mines.get((i, j), 0)  
 return c  
  
  
def open\_field(x, y, char):  
 if char == '💣':  
 color = 'red'  
 elif char == 1:  
 color = 'green'  
 elif char == 2:  
 color = 'lightblue'  
 elif char == 3:  
 color = 'blue'  
 elif char == 4:  
 color = 'brown'  
 elif char == "🚩":  
 color = 'orange'  
 else:  
 color = 'yellow'  
  
 draw\_cell(x, y, char, color)  
  
  
draw\_field()  
  
while GAME:  
 for e in event.get():  
 if e.type == QUIT:  
 exit()  
 pressed = mouse.get\_pressed()  
 if pressed[0] or pressed[2]:  
 x, y = mouse.get\_pos()  
 x, y = x // 50 \* 50 + 5, y // 50 \* 50 + 5  
 char = control\_field\_dict.get((x, y), 0)  
 if pressed[2]:  
 char = "🚩"  
 open\_field(x, y, char)