

Actividad 09 – QScene

Valencia frías Jonathan ricardo

Seminario de resolución de problemas de algoritmia

Lineamientos de evaluación

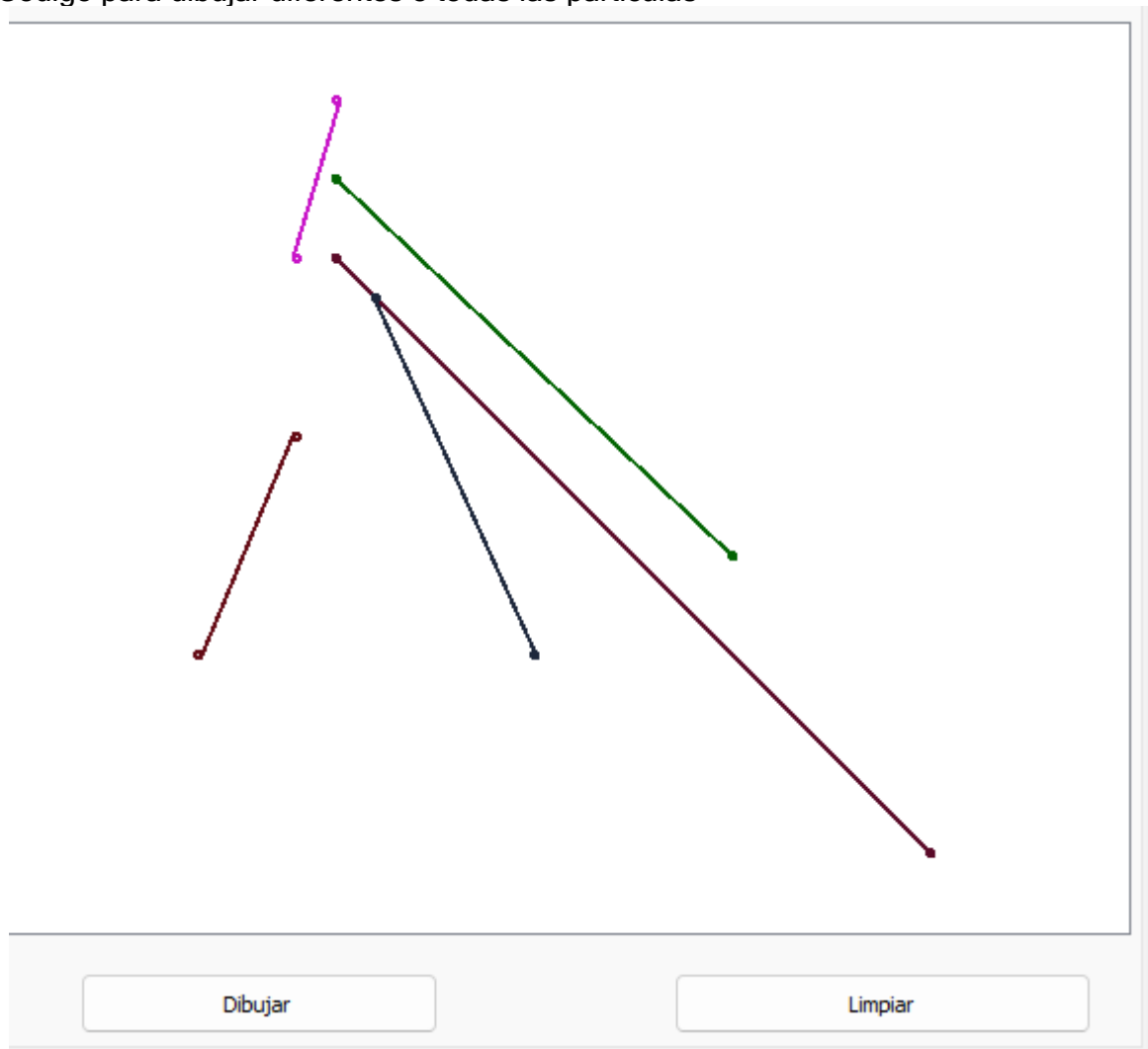
- El programa corre sin errores.
- El programa cuenta con todas las funciones
- Es capaz de realizar las tareas pedidas

Desarrollo

```
@Slot()
def dibujar(self):
    pen = QPen()
    pen.setWidth(2)

    for partícula in self.lista_particulas:
        r = partícula.red
        g = partícula.green
        b = partícula.blue
        color = QColor(r,g,b)
        pen.setColor(color)
        self.scene.addEllipse(partícula.origen_x,partícula.origen_y,3,3,pen)
        self.scene.addEllipse(partícula.destino_x,partícula.destino_y,3,3,pen)
        self.scene.addLine(partícula.origen_x+3,partícula.origen_y+3,partícula.destino_x,partícula.destino_y+3,pen)
```

Código para dibujar diferentes o todas las partículas



Escena con las partículas dibujadas

Conclusiones

En conclusión, se pudo conectar de manera correcta la interfaz con el proceso lógico de las clases

Referencias

<https://docs.python.org/3/>

Código

```
from wsgiref import headers
from PySide2.QtCore import Slot
from PySide2.QtGui import QPen, QColor, QTransform
from PySide2.QtWidgets import
QMainWindow, QFileDialog, QMessageBox, QTableWidgetItem, QGraphicsScene
from ui_mainwindow import Ui_MainWindow
from lista_particulas import Lista_Particulas
from _particula import Particula

class MainWindow(QMainWindow):
    def __init__(self):
        super(MainWindow, self).__init__()
        self.lista_particulas = Lista_Particulas()
        self.ui = Ui_MainWindow()
        self.ui.setupUi(self)
        self.ui.btn_inicio.clicked.connect(self.click_agregar_inicio)
        self.ui.btn_enviar_final.clicked.connect(self.click_agregar_fina
1)

        self.ui.mostrar_datos.clicked.connect(self.mostrar_datos)
        self.ui.actionAbrir.triggered.connect(self.accion_abrir_archivo)
        self.ui.actionGuardar.triggered.connect(self.accion_guardar_arch
ivo)

        self.ui.buscar_button.clicked.connect(self.buscar_particula)
        self.ui.mostrar_button.clicked.connect(self.mostrar_lista)
        self.ui.Dibujar.clicked.connect(self.dibujar)
        #self.ui.Limpiar.clicked.connect(self.mostrar_lista)
        self.scene = QGraphicsScene()
        self.ui.Escena_2.setScene(self.scene)

    @Slot()
    def dibujar(self):
        pen = QPen()
        pen.setWidth(2)

        for particula in self.lista_particulas:
            r = particula.red
```

```

        g = particula.green
        b = particula.blue
        color = QColor(r,g,b)
        pen.setColor(color)
        self.scene.addEllipse(particula.origen_x,particula.origen_y,
3,3,pen)
        self.scene.addEllipse(particula.destino_x,particula.destino_
y,3,3,pen)
        self.scene.addLine(particula.origen_x+3,particula.origen_y+3
,particula.destino_x,particula.destino_y,pen)

@Slot()
def Limpiar(self):
    self.scene.clear()

```