

Actividad 05 – clases y objetos

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

```
(2,0.4,0.4,0.6,0.4,0.1,6,5,4,0.28284271247461895)
(3,0.4,0.4,0.6,0.4,0.1,9,8,7,0.28284271247461895)
(0,0.2,0.2,0.4,0.2,0.1,3,2,1,0.28284271247461906)
(1,0.3,0.3,0.5,0.3,0.1,4,3,2,0.28284271247461906)
Press any key to continue . . .
```

Datos mostrados por la función mostrar

```
particula_1 = Particula(0,0.2,0.2,0.4,0.4,0.1,1,2,3)
particula_2 = Particula(1,0.3,0.3,0.5,0.5,0.1,2,3,4)
particula_3 = Particula(2,0.4,0.4,0.6,0.6,0.1,4,5,6)
particula_4 = Particula(3,0.4,0.4,0.6,0.6,0.1,7,8,9)
```

Creación de las partículas

```
lista.insertar_inicio(particula_4)
lista.insertar_inicio(particula_3)
lista.insertar_final(particula_1)
lista.insertar_final(particula_2)
```

Ingreso de las partículas

Conclusiones

En conclusión en conclusión pudimos conectar las diferentes clases necesarias para la actividad

Referencias

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

Código

```
import math
def distancia_euclidiana(x1,x2,y1,y2):
    return math.sqrt(((x2-x1)**2)+((y2-y1)**2))

from _particula import Particula

class Lista_Partículas:
    lt_particula=list()

    def insertar_inicio(self,p):
        self.lt_particula.insert(0,p)
    def insertar_final(self,p):
        self.lt_particula.append(p)
    def mostrar(self):
        for i in range(len(self.lt_particula)):
            print(str(self.lt_particula[i])+"\n")

from algoritmos import distancia_euclidiana

class Particula:
    id=0
    origen_x=0
    origen_y=0
    destino_x=0
    destino_y=0
    velocidad=0
    red=0
    green=0
    blue=0
    distancia=0.0

    def
__init__(self,id,origen_x,origen_y,destino_x,destino_y,velocidad,red,green,blue):
    self.id=id
    self.origen_x=origen_x
    self.origen_y=origen_y
    self.destino_x=destino_x
    self.destino_y=destino_y
    self.velocidad=velocidad
    self.red=red
    self.green=green
    self.blue=blue
    self.distancia=distancia_euclidiana(origen_x,destino_x,origen_y,destino_y)

    def __str__(self) -> str:
        return
        "("+str(self.id)+"," +str(self.origen_x)+"," +str(self.origen_y)+"," +str(self.destino_x
)+"," +str(self.origen_y)+"," +str(self.velocidad)+"," +str(self.blue)+"," +str(self.gree
n)+"," +str(self.red)+"," +str(self.distancia)+")"
```