

Documentación - Tarea 1

Equipo-2:

Alvaro Carrasco Escobar – 201873018-1

Camilo Contreras – 201873063-7

Nicolas Verdugo 202012033-1

Sebastián Álvarez Avendaño 202030018-6

Supuestos:

- Suponemos que la entrada de datos no cometerá errores, tales como “d0 62”.
- Suponemos para el stage 4, que las zonas son predeterminadas, o sea en la zona0 irá solo la puerta principal, la zona1 las demás puertas y ventanas, finalmente la zona2 de puros PIR.

Consideraciones:

- Para el movimiento de personas, decidimos no utilizar las flechas, en cambio utilizamos *w = arriba, s = abajo, a = izquierda, d = derecha*.
- En la Stage 2,3,4 en vez de hacer sonar la alarma con el archivo siren.wav, printeamos la alarma (wiu wiu wiu), pero aun así se dejó comentado las líneas de código para hacer la alarma sonar.



UNIVERSIDAD TECNICA
FEDERICO SANTA MARIA

Diseño y Programación
Orientados a Objetos

Profesor: AGUSTIN JOSE
GONZALEZ VALENZUELA

Solución:

Para la solución final, creamos tres listas con sensores, dependiendo de la zona que correspondan. Todas las diferentes clases interactúan:

La central posee una clase sirena, la cual nos sirve para tocar y parar la sirena dependiendo de las otras clases. La clase persona y PIR interactúan directamente, así podemos obtener el ángulo entre ellas y hacer los cálculos para ver si una persona invade un PIR.

En la central poseemos las listas con las clases puertas, ventanas y PIR, así la central se encarga de revisar todos los sensores de estas para determinar si uno está abierto o no, dependiendo si la alarma esta armada.

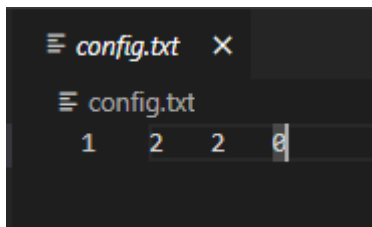
Las clases Door y Window poseen MagneticSensors, la cual es una clase que se extiende de Sensors, por lo que hereda sus propiedades. Los PIR, por otro lado, instancian una clase Sensor como tal.

La clase Stage 4, contiene el Main, es aquí donde ocurre toda la lógica, se instancian las otras clases y genera la interfaz para el usuario.

Stage 1:

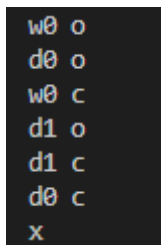
Config.txt:

2 puertas 2 ventanas 0 PIR



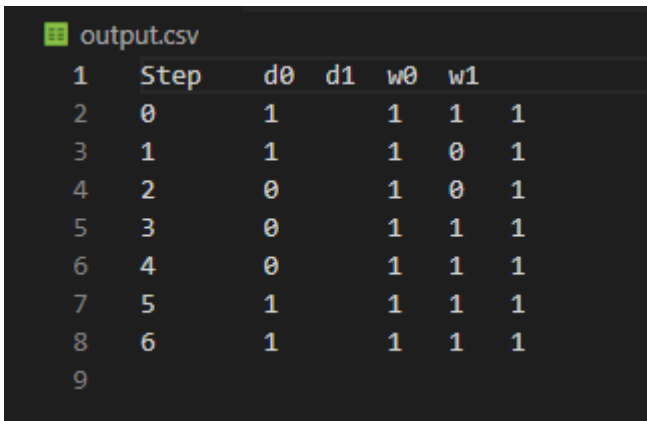
Input:

-Abrimos las 2 puertas y la primera venta, luego cerramos todas.



Output:

-Se ve claramente en el output como se van abriendo/cerrando las puertas y ventanas.



1	Step	d0	d1	w0	w1
2	0	1		1	1
3	1	1		1	0
4	2	0		1	0
5	3	0		1	1
6	4	0		1	1
7	5	1		1	1
8	6	1		1	1
9					

Stage 2:

Config.txt:

```
config.txt
1 2 2 0
2 siren.wav
3
```

Input:

-Abrimos la puerta 0 y la cerramos sin tener la central armada por lo que no suena la alarma, luego al armar la central, abrimos la puerta 0 -> suena la alarma, abrimos la puerta1 -> suena la alarma. Cerramos las 2 puertas para que deje de sonar la alarma, y finalmente desarmamos la central, para volver a abrir una puerta sin que suene la alarma.

```
d0 o
d0 c
k a
d0 o
Wiu wiu wiu wiu wiu ...
d1 o
Wiu wiu wiu wiu wiu ...
d0 c
Wiu wiu wiu wiu wiu ...
d1 c
k d
d1 o
x
```

Output:



UNIVERSIDAD TECNICA
FEDERICO SANTA MARIA

Diseño y Programación
Orientados a Objetos

Profesor: AGUSTIN JOSE
GONZALEZ VALENZUELA

-Se ve evidencia de lo mencionado arriba, el cómo se abren y cierran las ventanas, pero lo importante en esta stage era el comportamiento de la central y siren, como vemos mientras la central esta activa en step 5 hasta 9, cuando hay un 0 en puertas y ventanas, la sirena suena.

	Step	d0	d1	w0	w1	Siren	Central
1	Step	1	1	1	1	0	0
2	Step	0	1	1	1	0	0
3	Step	1	1	1	1	0	0
4	Step	1	1	1	1	0	1
5	Step	0	1	1	1	1	1
6	Step	0	0	1	1	1	1
7	Step	1	0	1	1	1	1
8	Step	1	1	1	1	0	1
9	Step	1	1	1	1	0	0
10	Step	1	0	1	1	0	0
11	Step	1	0	1	1	0	0
12							

Stage 3:

Config.txt:

2 puertas 1 ventanas 1 PIR



```
≡ config.txt
1 2 1 1
2 0.0 0.0 45 60 5
3 siren.wav
4
```

Input:

- Armamos la central con “k a”, luego creamos una persona en la coordenada 2.0 0.0
- Ya que no está en el rango del Pir, hacemos dos veces un movimiento para adelante con tal de que lo detecte el PIR y suena la alarma, “wiu wiu wiu wiu”, Luego alejamos a la persona, para que deje de sonar la alarma, y finalmente lo volvemos a entrar en el rango del PIR.

```
k a
c 2.0 0.0
p0 w
p0 w
Wiu wiu wiu wiu wiu ...
p0 w
Wiu wiu wiu wiu wiu ...
p0 s
Wiu wiu wiu wiu wiu ...
p0 s
p0 w
Wiu wiu wiu wiu wiu ...
p0 s
x
```

Output:

Se ve claramente en el output que en el step 1 activamos la central, luego agregamos la persona y en el step4, esta persona entra en el rango de detección del PIR, en el step 9 se va del rango, y vuelve a entrar en el step 10.



UNIVERSIDAD TECNICA
FEDERICO SANTA MARIA

Diseño y Programación
Orientados a Objetos

Profesor: AGUSTIN JOSE
GONZALEZ VALENZUELA

1	Step	d0	d1	w0	Pir0	Siren	Central		
2	0	1	1	1	1	0	0		
3	1	1	1	1	1	0	1		
4	2	1	1	1	1	0	1	2.0	0.0
5	3	1	1	1	1	0	1	2.0	0.5
6	4	1	1	1	0	1	1	2.0	1.0
7	5	1	1	1	0	1	1	2.0	1.5
8	6	1	1	1	0	1	1	2.0	1.0
9	7	1	1	1	1	0	1	2.0	0.5
10	8	1	1	1	0	1	1	2.0	1.0
11	9	1	1	1	1	0	1	2.0	0.5
12									

Stage 4:

Config.txt:

2 puertas 1 ventanas 3 PIR



UNIVERSIDAD TECNICA
FEDERICO SANTA MARIA

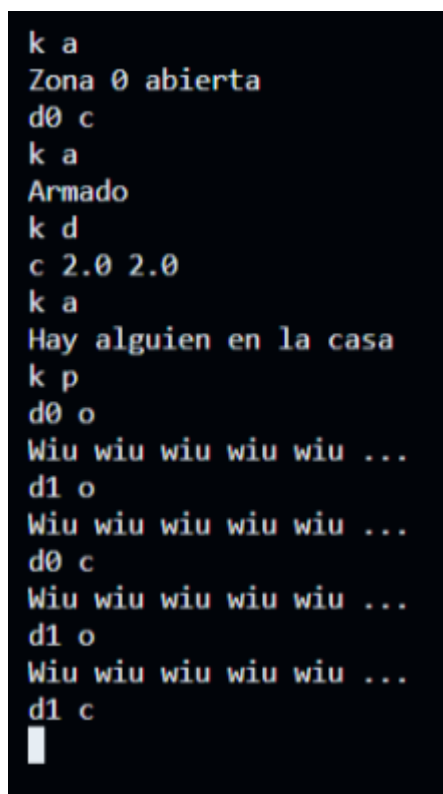
Diseño y Programación
Orientados a Objetos

Profesor: AGUSTIN JOSE
GONZALEZ VALENZUELA



Input:

-Probamos abrir y cerrar puertas tanto con perímetro y área armado.



Output:



1	Step	d0	d1	w0	Pir0	Pir1	Pir2	Siren	Central
2	0	1	1	1	1	1	0	0	
3	1	0	1	1	1	1	0	0	
4	2	0	1	1	1	1	0	0	
5	3	1	1	1	1	1	0	0	
6	4	1	1	1	1	1	0	1	
7	5	1	1	1	1	1	0	0	
8	6	1	1	1	1	0	1	0	2.0 2.0
9	7	1	1	1	1	0	1	0	2.0 2.0
10	8	1	1	1	1	0	1	0	2.0 2.0
11	9	0	1	1	1	0	1	1	2.0 2.0
12	10	0	0	1	1	0	1	1	2.0 2.0
13	11	1	0	1	1	0	1	1	2.0 2.0
14	12	1	0	1	1	0	1	1	2.0 2.0
15	13	1	1	1	1	0	1	0	2.0 2.0
16									

3 dificultades encontradas:

- Problemas en la Stage 4 con el tema de los ángulos y poder detectar exactamente si la persona estaba en el Área del PIR.
- En la Stage 2 tuvimos problemas al actualizar valores de Central entre 1 y 0 o Siren 1 y 0 etc.
- Problemas al utilizar el sonido entregado de siren.wav.

UML:

