

a)		15 puntos						
Global	funcion							
a	x	y	z	l	i			
[1, 4, 8]						1 pto		
	-2					2 ptos	<i>Pueden ir en una misma fila.</i>	
		3						
			[1, 4, 8]					
				[]		1 pto		
	1					2 ptos		
					0			
				[1]				
					1			
				[1, 3]		3 ptos		
				[1, 3, 4]				
					2	2 ptos		
				[1, 3, 4, 8]				
[1, 3, 4, 8]						1 pto		
	[1, 3, 4, 8]					3 pts	<i>Considerar los prints solo si demuestra saber hacer el ruteo.</i>	
1 b) Análisis de algoritmos								
El algoritmo inserta un elemento en la posición x, en referencia a una lista dada						5 ptos		

Pregunta 2:		40 pts			
Item	0 Pto	3 Ptos	6Ptos	10 Ptos	14 Ptos
Manejo de estructuras de datos	No demuestra tener noción de cómo usar las estructuras de datos	Al menos en una función utiliza de forma completamente correcta la estructura. Comete errores de uso de los métodos propios de cada estructura	Usa correctamente las estructuras de datos en las 3 funciones, sin cometer errores		
Función menor_publico()	No demuestra nociones de cómo resolver el problema	Demuestra tener una noción de qué hacer, tiene la lógica de la solución pero comete a los más 2 errores en el proceso. Ej: No retorna una tupla, no recorre de forma correcta el conjunto	Comete algún error importante o 2 errores leves. Ej: Sólo cambia el valor del menor público pero no guarda el nombre de la ciudad, recorre por posición el conjunto, retorna la tupla en un orden incorrecto, es decir entrega el público y después la ciudad, mal inicializado el valor del menor.	Retorna correctamente el resultado esperado.	
Función gira()	No demuestra nociones de cómo resolver el problema	Demuestra tener una noción de qué hacer, tiene la lógica de la solución pero con a lo más 2 errores en el proceso. Ej: no inicializa las listas, mal uso del sort(), no recorre bien el conjunto, agrega más de un parámetro en el método append().	Comete algún error importante o 2 errores leves. Ej: ordena de forma incorrecta, es decir de mayor a menor, entrega el público en vez de la ciudad	Retorna correctamente el resultado esperado	
Función nueva_gira()	No demuestra nociones de cómo resolver el problema	Demuestra tener noción de qué hacer, se aprecia lógica en el desarrollo pero comete errores importantes en su implementación. Ej: recorre mal el diccionario, no retorna la lista ordenada de forma ascendente según su capacidad de público.	Retorna el resultado esperado, pero en su desarrollo cometió varios errores. Ej: llama incorrectamente a la función gira, recorre el diccionario sin especificar de qué forma lo recorre (sin usar .items()), entrega de la lista ordenada ascendente según su capacidad, calculo incorrecto de la recaudación.	Retorna el resultado esperado pero cometió un pequeño error en su desarrollo, como por ejemplo pasar más de un valor en el método append, no inicializa la lista.	Retorna correctamente el resultado esperado
Descuentos		0 Pto			
Forma y expresiones	No comete errores graves con respecto al lenguaje; a lo más errores atribuibles más a descuidos que ignorancia (por ejemplo, en una ocasión olvidar los dos puntos al final de la línea, confundir = y == o confundir () y [] o , ocupa ≤ en lugar de <=, pero el resto de las veces lo hace bien)		-4 Puntos		
			Demuestra no entender construcciones importantes del lenguaje (por ejemplo: no indenta, le pone una condición al else, confunde el string x con la variable x, asigna valor a variables de forma incorrecta 4=x), consistentemente no le pone los paréntesis finales a d.items() o d.values(). Un descuento de -4 por tipo de error.		

Pregunta 3: 40 pts				
Item	0 Pto	5 Ptos	8 Ptos	10 Ptos
Acceder a cada enemigo	No demuestra nociones de cómo realizar esta acción o los errores son mayores a los aciertos.	Algorítmicamente sabe como resolver esta acción pero comete 2 errores en el proceso. Ej: abre el archivo con el nombre del ejemplo y no el parámetro, mal uso del split al separar la línea, abre y cierra en la posición incorrecta del código el archivo.	Resuelve esta acción pero comete algún error menor en el proceso. Ej: olvida cerrar el archivo, abre el archivo en un modo de no-lectura, olvida el strip.	Recorre el archivo entregado como parámetro y separa los datos de la línea correctamente.
Radio de acción de la bomba	No demuestra nociones de cómo realizar esta acción o los errores son mayores a los aciertos.	Algorítmicamente demuestra saber como identificar si una unidad es destruida o no, pero en el proceso comete 2 errores. Ej: no separa la posición leída desde el archivo por la coma, calcula de forma incorrecta la distancia entre dos puntos.	Determina si una unidad es destruida o no pero comete algún error menor en el proceso. Ej: olvida pasar la posición x.y leía del archivo a entero, no incluye unidades a una distancia igual al radio.	Resuelve correctamente esta acción.
Diccionario resultados.	No demuestra nociones de cómo realizar esta acción o los errores son mayores a los aciertos.	Algorítmicamente resuelve este ítem, pero comete 2 errores en el proceso. Ej: cuenta la destrucción de todas las clases y no sólo las de la lista, agrega de forma incorrecta la llave al diccionario, cuenta de forma equivocada, ejemplo apariciones de la clase y no las destrucciones.	Resuelve el problema pero comete algún error menor en el proceso. Ej: olvida inicializar el diccionario vacío, no inicializa el contador en 0 o lo hace de forma incorrecta.	Resuelve correctamente esta acción.
Actualización archivo	No demuestra nociones de cómo realizar esta acción o los errores son mayores a los aciertos.	Algorítmicamente resuelve este punto, pero comete a lo más 2 errores en el proceso. Ej: intenta leer y escribir en el archivo a la vez, mal uso de los métodos de apertura, no ocupa una estructura o archivo temporal.	Resuelve el problema con archivo temporal o una lista pero comete algún error menor en el proceso. Ej: no cierra los archivos, crea el temporal pero no devuelve el contenido al archivo original, escribe de forma incorrecta en el archivo, olvida el '\n'	Resuelve correctamente esta acción.
Descuentos		0 Pto	-4 Puntos	
Sintaxis y expresiones	No comete errores graves con respecto al lenguaje; a lo más errores atribuibles más a descuidos que ignorancia (por ejemplo, en una ocasión olvidar los dos puntos al final de la línea, confunde = y == o confunde () y [] o , ocupa ≤ en lugar de <=, pero el resto de las veces lo hace bien)		Demuestra no entender construcciones importantes del lenguaje (por ejemplo: no indenta, le pone una condición al else, hace while con rango, confunde el string x con la variable x, asigna valor a variables de forma incorrecta 4=x). Un descuento de -4 por tipo de error.	