

Pregunta 2: 40 pts				
Item	0 Pto	4 Ptos	7 Ptos	11 Ptos
Uso de estructuras de datos	No demuestra saber como manejar las estructuras de datos del problema	Demuestra saber como manejar las estructuras de datos utilizadas al menos en una función.	Demuestra uso impecable de las estructuras de datos en todas las funciones.	
Función agregar_medicion	No demuestra nociones de cómo resolver el problema	La función tiene la lógica descrita pero comete 1 o más errores. Ej: recorre la lista pero de forma incorrecta, no sabe donde agregar la nueva medición, no sabe como crea la tupla a agregar, ocupa sort para reordenar el registro.	La lógica de la solución está correcta pero comete algún error menor. Ej: no sabe ocupar el método insert, agrega correctamente pero agrega la medición varias veces, no considera el caso de agregar el valor al final de la lista, inserta valores por separado en vez de una tupla.	La función está correcta completamente.
Función corregir_medicion	No demuestra nociones de cómo resolver el problema	La función tiene la lógica descrita pero comete 1 o más errores. Ej: recorre la lista pero de forma incorrecta, no sabe donde agregar la medición corregida, modifica la tupla original, ocupa sort para reordenar el registro.	La lógica de la solución está correcta pero comete algún error menor. Ej: reemplaza la tupla en la posición de la medición anterior, si ocupa la función anterior lo hace de forma equivocada, elimina la tupla a modificar pero de forma errónea.	La función está correcta completamente.
Función sensores_sobre_umbral	No demuestra nociones de cómo resolver el problema	La función tiene la lógica descrita pero comete 1 o más errores. Ej: recorre la lista pero de forma incorrecta, no sabe desempaquetar valores o acceder a los valores internos de la tupla.	La lógica de la solución está correcta pero comete algún error menor. Ej: agrega elementos repetidos a la lista, filtra de forma incorrecta los valores sobre u.	La función está correcta completamente.
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.		

Pregunta 3: 40 pts					
Item	0 Pto	4 Ptos	7 Ptos	10 Ptos	12 Ptos
Uso de estructuras de datos	No demuestra saber como manejar las estructuras de datos del problema	Demuestra saber como recorrer o acceder a los datos en una de las estructuras de datos utilizadas.	Demuestra uso impecable de las estructuras de datos.		
Cálculo de las distancias	No demuestra saber como calcular la distancia entre los destinos y la empresa	Realiza el calculo de la distancia entre la empresa y todos los destinos pero comete algún error leve en su implementación. Ej: eleva a 1/2 en división entera.	Calcula de forma impecable las distancias entre la empresa y TODOS los destinos		
Filtra destinos alcanzables	No demuestra nociones de cómo obtener los viajes a realizar.	Demuestra tener nociones de cómo filtrar los destinos que se pueden realizar, pero comete 2 errores en la implementación. Ej: el cálculo de las distancias alcanzables no es correcto, realiza división entera cuando debió ser float.	Determina de forma impecable los destinos factibles de realizar por la empresa dada las restricciones existentes.		
Cálculo del ingreso por encomienda	No demuestra nociones de cómo obtener los ingresos por encomienda	Realiza el cálculo del Ingreso para las encomiendas, pero comete algún error en su implementación. Ej: no logra asociar para una misma valija sus dimensiones y ubicación.	Realiza correctamente el cálculo del Ingreso de las encomiendas		
Listar las encomiendas a realizar	No demuestra nociones de cómo generar la lista de los viajes a realizar en base a las limitantes de distancia e ingreso existentes	Genera una lista de encomiendas pero como 2 o 3 errores en el proceso. Ej: la lista no contiene los códigos de las valijas, o los tiene pero además otro dato no solicitado, no filtra las 10 valijas de mayor ingreso, repite encomiendas en el listado.	Genera el listado de encomiendas a realizar pero 2 errores leves en el proceso. Ej: asume que hay más de 10 encomiendas, repite encomiendas en la lista final. Si ocupa la función sort() o reverse(), asigna el resultado a una variable.	Genera el listado de encomiendas pero comete algún error menor. Ej: entrega las 10 encomiendas que MENOS ingresos generan.	Entrega el listado de las encomiendas que se pueden realizar en los últimos 30 minutos y que generan los mayores ingresos de forma correcta. Considera el caso en que no existan 10 encomiendas
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.		