

1.a)			
def prox_venc(cuentas, fecha):			Observación:
dest = open('vence.txt', 'w')	2	ptos	Una línea esta correcta si está en la posición correcta (funcionalmente) y con la indentación correcta.
arch = open(cuentas)			
m, a = fecha.split('/')	1	pto	
for li in arch:	1	pto	
datos = li.strip().split(':')	1	pto	
f = datos[-1].split('/')	1	pto	
mm, aa = f[1], f[-1]	1	pto	
if mm == m and aa == a:	1	pto	
dest.write(datos[0] + '\n')	2	pto	
arch.close()	2	ptos	
dest.close()			
1.b)			
Retorna contraseñas seguras, aquellas con al menos un dígito y un largo mayor al pasado como parámetro (mínimo)	8	ptos	
<i>Explicaciones donde no explican lo que es una contraseña segura, o lo hace parcialmente correcto.</i>	4	ptos	

Pregunta 2:		40 pts			
Item	0 Pto	3 Ptos	6 Ptos	10 Ptos	12 Ptos
Manejo de archivos	No demuestra tener conocimiento sobre el manejo de archivos, nunca cierra los archivos que abre.	Al menos en una función utiliza de forma completamente correcta archivos. No usa el nombre del archivo que se pasa como parámetro, deja abierto los archivos.	Usa correctamente los archivos, al menos en 2 de las 3 funciones solicitadas. Abre y cierra los archivos donde corresponde.		
Función Leer_pukamones()	No demuestra nociones de cómo resolver el problema o los errores son mayores a los aciertos	Demuestra saber como resolver el problema, pero usa mal el split(), no crea y/o manipula la lista de forma incorrecta, y/o hace el return dentro del ciclo agrega más de un elemento por medio del append	Demuestra tener conocimiento de saber cómo resolver el problema, pero comete errores importantes en su implementación, como no convertir la ubicación a número o no crear de forma correcta la tupla.	Implementa correctamente la función, realiza la conversión de str a int y abre y cierra correctamente los archivos	
Función Calcular_cercanía()	No demuestra nociones de cómo resolver el problema o los errores son mayores a los aciertos	Demuestra tener nociones de cómo resolver el problema, pero comete muchos errores en su implementación, o hace el return dentro del ciclo o no agrega los nombres de los pukamones en una lista.	Demuestra tener claro cómo resolver el problema, pero comete errores en su implementación, como por ejemplo no retorna el diccionario, errores en la categorización de la cercanía de los pukamones, comete errores al agregar los pukamones a la lista del diccionario, comete un error al llamar a la función anterior, entre otros.	Demuestra tener claro cómo resolver el problema, pero comete errores leves en su implementación, como por ejemplo usa el nombre del archivo del ejemplo y no el parámetro.	Implementa correctamente la función. crea sin errores el diccionario y categoriza sin problemas a los pukamones según su grado de cercanía
Función itinerario_pukamones()	No demuestra nociones de cómo resolver el problema o los errores son mayores a los aciertos	Demuestra tener nociones de cómo resolver el problema pero comete muchos errores en su implementación, como por ejemplo llama de forma incorrecta a la(s) función(es) anteriores, abre los archivos del modo incorrecto borrando toda la información de los archivos.	Demuestra tener claro cómo resolver el problema, pero comete errores en su implementación, como por ejemplo llama de forma incorrecta a la(s) función(es) anteriores, no tiene cuidado con los tipos de datos al momento de escribir.	Demuestra tener claro cómo resolver el problema, pero comete errores leves en su implementación, como por ejemplo usa print en vez de WRITE, usa los nombres de los archivos y no los parámetros.	Implementa de forma correcta la función.
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 ptos		
Item	0 Pto	5 Ptos	10 Ptos	15 Ptos
Crea el loop para leer los archivos con la evaluaciones	No realiza un loop para leer diferentes archivos de notas.	Realiza un loop para leer diferentes archivos de notas.		
Obtiene todos los datos a actualizar en un Resumen Académico	No obtiene ningún dato para actualizar la información	Obtiene el código del curso o los datos de un alumno, pero no ambos.	Obtiene el código del curso, los datos del alumno y además los parsea correctamente	
Calcula correctamente el nuevo promedio	No calcula el promedio	Calcula el promedio, pero contabiliza erradamente el nuevo curso.	Calcula correctamente el promedio	
Genera de forma correcta el nuevo archivo Insertando la nueva información en el lugar que corresponde	No escribe el nuevo curso o utiliza la función sort() en este proceso.	Escribe el nuevo curso en cualquier posición o lo inserta más de 2 veces en algunos casos.	Escribe el curso en el semestre que corresponde, pero no concidera casos como la pre-existencia de dicho curso o que no hayan cursos de semestres posteriores a éste.	No comete errores al escribir el curso en el semestre que corresponde
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 olivdar 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). Comete más de un error en la lectura/escritura de archivos.	