

<b>1.a)</b>	<b>25 puntos</b>	
def empresas(post):		<b>Observación:</b>
arch_P = open(post)	2 ptos	Una línea está correcta si está en la posición correcta (funcionalmente) y con la indentación correcta.
emp = list()		
for li in arch_P:	1 pto	
r, p, e = li.strip().split('#')	1 pto	
if e not in emp:	1 pto	
emp.append(e)	1 pto	
arch_P.close()	1 pto	
return emp	1 pto	
<b>1.b) Opción 1</b>		<b>1.b) Opción 2</b>
def registros(tit, post):		def registros(tit, post):
emp = empresas(post)	1 pto	li = '{0};{1};{2}\n'
li = '{0};{1};{2}\n'	1 pto	emp = empresas(post)
for e in emp:	1 pto	for e in emp:
arch_E = open(e + '.txt', 'w')	2 ptos	arch_E = open(e + '.txt', 'w')
arch_P = open(post)		arch_T = open(tit)
for pos in arch_P:	1 pto	for titu in arch_T:
r, p, e2 = pos.strip().split('#')	1 pto	n, r2 = titu.strip().split(';')
if e2 == e:	1 pto	arch_P = open(post)
arch_T = open(tit)	1 pto	for pos in arch_P:
for titu in arch_T:	1 pto	r, p, e2 = pos.strip().split('#')
n, r2 = titu.strip().split(';')	1 pto	if r2 == r:
if r2 == r:	1 pto	if e2 == e:
arch_E.write(li.format(r, n, p))	2 ptos	arch_E.write(li.format(r, n, p))
arch_T.close()		arch_P.close()
arch_E.close()	2 ptos	arch_E.close()
arch_P.close()		arch_T.close()
return None	1 pto	return None

<b>Pregunta 2:</b>		<b>35 ptos</b>		
<b>Item</b>	<b>0 Pto</b>	<b>5 Ptos</b>	<b>8 Ptos</b>	<b>10 Ptos</b>
Manejo de archivos	No demuestra entender el uso de archivos	Demuestra entender como utilizar archivos, abre-lee-cierra donde y cómo corresponde.		
Funcion a)	No demuestra tener la noción de cómo resolver el problema	Algorítmicamente el desarrollo tiene sentido, cubriendo todos los puntos de la pregunta. Los errores cometidos son por falta de manejo del lenguaje.	El problema fue resuelto pero con algún error en la implementación, ej: no agrega correctamente elementos al diccionario.	Resuelve el problema correctamente
Funcion b)	No demuestra tener la noción de cómo resolver el problema	Algorítmicamente el desarrollo tiene sentido, cubriendo todos los puntos de la pregunta. Los errores cometidos son por falta de manejo del lenguaje.	El problema fue resuelto pero con algún error en la implementación, ej: abre directamente el archivo clientes.txt en vez de utilizar el parámetro de la función.	Resuelve el problema correctamente
Funcion c)	No demuestra tener la noción de cómo resolver el problema	Algorítmicamente el desarrollo tiene sentido, cubriendo todos los puntos de la pregunta. Los errores cometidos son por falta de manejo del lenguaje.	El problema fue resuelto pero con algún error en la implementación, ej: olvida inicializar el contador en 0 para cada clase. Asume que las clases disponibles son sólo las del ejemplo.	Resuelve el problema correctamente
<b>Descuentos</b>	<b>0 Pto</b>	<b>-4 Puntos</b>		
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	5 Ptos	8 Ptos	10 Ptos	12 Ptos
Manejo de archivos	No demuestra entender el uso de archivos	Demuestra entender como utilizar archivos en almenos una función, abre-lee-cierra donde y cómo corresponde.	Demuestra entender como utilizar archivos en todas las funciones, abre-lee/escribe-cierra donde y cómo corresponde. No equivoca/confunde los modos de acceso.		
Funcion a)	No demuestra tener la noción de cómo resolver el problema	Algorítmicamente el desarrollo tiene sentido, cubriendo todos los puntos de la pregunta. Comete algún error por falta de manejo del lenguaje.	Resuelve el problema correctamente		
Funcion b)	No demuestra tener la noción de cómo resolver el problema	Algorítmicamente el desarrollo tiene sentido, cubriendo todos los puntos de la pregunta. Los errores cometidos son por falta de manejo del lenguaje.	Resuelve algorítmicamente el problema, pero comente más de 2 omisiones o errores importante como: abre/cierra los archivos donde no corresponde, olvida el salto de línea al escribir, trata de modificar el archivo a la vez que lee en él.	Resuelve algorítmicamente el problema, pero comente algún error leve, ej: ocupa el archivo clientes.txt en vez del parámetro de la función, olvida agregar el salto de línea al final de la línea.	Resuelve el problema correctamente
Funcion c)	No demuestra tener la noción de cómo resolver el problema	Algorítmicamente el desarrollo tiene sentido, cubriendo todos los puntos de la pregunta. Los errores cometidos son por falta de manejo del lenguaje.	Resuelve algorítmicamente el problema, pero comente más de 2 omisiones o errores importante como: no llama el archivo con el nombre indicado, ingresa la clase en la linea	Resuelve algorítmicamente el problema, pero comente algún error leve, ej: ocupa el archivo clientes.txt en vez del parámetro de la función, olvida agregar el salto de línea al final de la línea.	Resuelve el problema correctamente
<b>Descuentos</b>	<b>0 Pto</b>	<b>-4 Puntos</b>			
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 olviddar 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.			