

9.4-Fitxers

1) El fitxer `qualificacions.csv` conté les qualificacions d'un curs. Durant el curs es van fer dos exàmens parcials de teoria i un examen de pràctiques. Els alumnes que van tenir menys de 4 en algun d'aquests exàmens van poder repetir-ho al final del curs (convocatòria ordinària). Escriure un programa que contingui les funcions següents:

1. Una funció que rebí el fitxer de qualificacions i torni una llista de diccionaris on cada diccionari conté la informació dels exàmens i l'assistència d'un alumne. La llista ha d'estar ordenada pels cognoms.
2. Una funció que rebí una llista de diccionaris com la que torna la funció anterior i afegeixi a cada diccionari un nou parell amb la nota final del curs. El pes de cada parcial de teoria a la nota final és d'un 30%, mentre que el pes de l'examen de pràctiques és d'un 40%.
3. Una funció que rebí una llista de diccionaris com la que torna la funció anterior i torni dues llistes, una amb els alumnes aprovats i una altra amb els alumnes suspensos. Per aprovar el curs, l'assistència ha de ser més gran o igual que el 75%, la nota dels exàmens parcials i de pràctiques més gran o igual que 4 i la nota final més gran o igual que 5.

def `nota(cifra)`:

```
cifra = cifra.replace(',','.')
return float(cifra)
```

def `calificaciones(ruta)`:

```
try:
    # Abrimos el fichero en modo lectura
    f = open(ruta, 'r')
except FileNotFoundError:

-----

    return
# Leemos el fichero por líneas en una lista
lineas = f.readlines()
# Cerramos el fichero
f.close()
# Leemos las claves del primer elemento de la lista, eliminamos el cambio de línea que aparece al final y dividimos la cadena
a por el punto y coma.
claves = lineas[0][:-1].split(";")
# Creamos la lista de calificaciones
calificaciones = []
# Recorremos las líneas del fichero y para cada línea creamos un diccionario que añadimos a la lista de calificaciones.
for i in lineas[1:]:
    # Eliminamos el cambio de línea del final y dividimos la cadena por el punto y coma.
    valores = i[:-1].split(";")
    # Creamos un diccionario vacío para ir añadiendo los datos de cada alumno.
    alumno = {}
    # Recorremos la lista de valores y los añadimos al diccionario.
    for j in range(len(valores)):
        alumno[claves[j]] = valores[j]
    # Añadimos el diccionario a la lista de calificaciones
    calificaciones.append(alumno)
return calificaciones
```

```
def añadir_nota_final(calificaciones):
```

```
    def nota_final(alumno):
```

```
        if alumno['Ordinario1']: #Si el alumno se ha presentado al examen de repesca del primer parcial tomamos esa nota como
la nota del primer parcial
            parcial1 = nota(alumno['Ordinario1'])
        elif alumno['Parcial1']:
            parcial1 = nota(alumno['Parcial1'])
        else: # No se ha presentado al primer parcial ni a la repesca en el ordinario
            parcial1 = 0
        if alumno['Ordinario2']: #Si el alumno se ha presentado al examen de repesca del segundo parcial tomamos esa nota com
o la nota del segundo parcial
            parcial2 = nota(alumno['Ordinario2'])
        elif alumno['Parcial2']:
            parcial2 = nota(alumno['Parcial2'])
        else: # No se ha presentado al segundo parcial ni a la repesca en el ordinario
            parcial2 = 0
        if alumno['OrdinarioPracticas']: #Si el alumno se ha presentado al examen de repesca de prácticas tomamos esa nota com
o la nota de prácticas
            practicas = nota(alumno['OrdinarioPracticas'])
        elif alumno['Practicas']:
            practicas = nota(alumno['Practicas'])
        else:
            practicas = 0
        alumno['Final1'] = parcial1
        alumno['Final2'] = parcial2
        alumno['FinalPracticas'] = practicas
        alumno['NotaFinal'] = parcial1 * 0.3 + parcial2 * 0.3 + practicas * 0.4
        return alumno

    # Aplicamos la función nota_final a todos los alumnos
    return list(map(nota_final, calificaciones))
```

```
def aprobados_suspensos(calificaciones):
```

```
    # Creamos las listas de aprobados y suspensos vacías
```

```
    aprobados = []
```

```
    suspensos = []
```

```
    # Recorremos los alumnos del curso
```

```
    for alumno in calificaciones:
```

```
        # Si se cumplen las condiciones para aprobar añadimos el nombre del alumno a la lista de aprobados y si no a la de suspensos.
```

```
        if all([int(alumno['Asistencia'][:1]) >= 75, alumno['Final1'] >= 4, alumno['Final2'] >= 4, alumno['FinalPracticas'] >= 4, alumno['NotaFinal'] >= 5]):
```

```
            aprobados.append(alumno['Apellidos'] + ', ' + alumno['Nombre'])
```

```
        else:
```

```
            suspensos.append(alumno['Apellidos'] + ', ' + alumno['Nombre'])
```

```
    return aprobados, suspensos
```

```
    # Llamada a las funciones de prueba
```

```
-----  
aprobados, suspensos = aprobados_suspensos(añadir_nota_final(calificaciones('calificaciones.csv')))
```

Pràctica:

Dissenya un programa diferent, enunciat inclòs, que combini els exemples anteriors, prova'l, mostra'l i enganxa'l aquí.