

| | | | |
|--|--|--|--|
| | | | |
|--|--|--|--|

APELLIDOS :

NOMBRE :

- Pon tu nombre en todas las hojas del examen.
- Resuelve cada problema en una hoja separada.
- Tiempo máximo: 1,5 horas.

1. (Total 1,0 punto) Disponemos de un fichero de texto que contiene en cada línea del fichero registros de tipo *NOTA_MEDIA* con el siguiente formato fijo:

APELLIDO1 APELLIDO2 NOMBRE NOTA_MEDIA

Por ejemplo:

| | | | |
|-----------|----------|------------------|-----|
| Abad | García | Alberto | 6.5 |
| Angulo | Espinosa | Aurora | 6.2 |
| ... | | | |
| Valencia | Alvarez | Claudio | 5.0 |
| Zarzalejo | Espinosa | María del Carmen | 4.5 |

El primer campo ocupa 30 caracteres, al igual que el segundo y el tercero. El cuarto campo, la nota media, contiene un número real entre 0.0 y 10.0.

- a) (0,5 puntos) Propón una implementación en C para este tipo de dato.

```
typedef struct {
    char ape1[30], ape2[30], nombre[30];
    float nota_media;
} NOTA_MEDIA;
```

- b) (0,5 puntos) Como ya se ha visto en la asignatura, en código algorítmico un registro de un fichero de tipo T se puede leer utilizando la orden: *leer_f(f, V)*; suponiendo que previamente se ha comprobado que no es fin de fichero mediante *fin_f(f, eof)*; donde:

f: fichero de T;
V: T;
eof: lógico;

En código C, no es posible aplicar este esquema directamente ya que sólo trata archivos de texto, no reconoce los registros y además existen distintas opciones de implementar *leer_f()* en función de qué se esté leyendo (*fscanf()*, *fgets()*, *getch()*) y que ya llevan su propia implementación de *fin_f()* incorporada.

Realiza una función en C que tenga en cuenta todos estos aspectos y que nos permita leer un registro de tipo *NOTA_MEDIA* del apartado anterior y que respete la siguiente especificación:

Algoritmo leer_registro_notas (f, N, ok) es
f: fichero de *NOTA_MEDIA*; {parámetro dato}
N: registro de *NOTA_MEDIA*; {parámetro dato-resultado}
ok: numérico; {parámetro resultado}

La variable ok debe indicar si se ha llegado a fin de fichero (valor -1), o si los datos son correctos (valor 1).

```
int leer_registro_notas (FILE * f, NOTA_MEDIA * nota)
{
    int resp=1;
    if (fscanf(f,"%30s%30s%30s%5.2f",nota->ape1,nota->ape2,nota->nombre,
        &(nota->nota_media))) == EOF)
        resp = -1;
    return resp;
}
```

2. (3 puntos) Ahora debes suponer que la función en C *leer_registro_notas()* del apartado anterior existe y se puede usar.

Realiza un programa en C que recorra todos los registros de un fichero de notas y nos indique por pantalla cuál ha sido la nota media más alta del fichero y a qué alumn@s correspondían esos datos.

Debes tener en cuenta que puede haber varios alumn@s que compartan la nota media máxima. En ese caso debes guardar en un vector los datos de todos ellos, hasta un máximo de 10. Si en un momento dado se encuentran más de 10 alumn@s con la misma nota igual al máximo, debes indicarlo, avisando del error y debe detenerse el recorrido del fichero.

Nota: ten en cuenta que puedes hacer este programa con independencia de cómo esté codificada la función *leer_registro_notas()*; sólo debes usarla, respetando sus especificaciones, pero no debes volver a programarla.

```
void muestra (NOTA_MEDIA nota) {
    printf("\nHe leído:\n%-30s%-30s\n%-30s\n", nota.ape1, nota.ape2, nota.nombre);
    printf("\t %5.2f\n", nota.nota_media);
}
```

```
void copia (NOTA_MEDIA origen, NOTA_MEDIA *destino) {
    strcpy(destino->ape1, origen.ape1);
    strcpy(destino->ape2, origen.ape2);
    strcpy(destino->nombre, origen.nombre);
    destino->nota_media = origen.nota_media;
}
```

```
main(){
    FILE * f;
    NOTA_MEDIA reg, maximos[10];
    int i, j=0, ok=1;
    float maxima=-1.0;
    f = fopen("notas.txt", "r");
    if (f != NULL) {
        while (leer_registro_notas(f, &reg) == 1 && ok) {
            if (reg.nota_media > maxima) {
                maxima = reg.nota_media;
                i = 0;
                copia (reg, &maximos[i]);
            } else
                if (reg.nota_media == maxima){
                    if (i < 10) {
                        i++;
                        copia(reg, &maximos[i]);
                    } else ok = 0;
                }
        }
    }
```

```
    }  
fclose(f);  
if (ok)  
    if (maxima >= 0.0) {  
        printf("\nLa nota media maxima es %f", maxima);  
        printf("\nObtenida por: ");  
        for (j=0; j <= i; j++) muestra(maximos[j]);  
    }  
    else  
        printf("\nError en la lectura, demasiados con la misma nota máxima");  
}  
system("PAUSE");  
}
```