



# **Título:** PROYECTO FINAL DE AÑO. GYMNASTIKSKO.

**Componentes del grupo:**

Sofía Guerrero Neto, 53999

Micaela Cristina Amaya López, 53814

**Fecha de entrega:** 14/03/2019

**Profesora:** Cecilia E. García Cena

**Grupo:** A104 - Ingeniería Electrónica Industrial y Automática

## ÍNDICE

1. Código en C .....	3
2. Código en Arduino .....	22
3. Memoria .....	28

## CÓDIGO EN C

```
#include<stdio.h>
#include<windows.h>
#include<malloc.h>
//Incluir las de Arduino
//#include "SerialPort.h"

#define NOM 25 //Se puede asignacion dinamica de memoria? -> Mejor no porque no le vas
a pedir la dimension al usuario

//Mirar la opcion de meter todo lo del main() en distintas funciones para evitar el problema
de opcperf==4
//y asi tambien ordenar un poco el codigo

//Flags para el case 3

//Estructura para los perfiles
typedef struct{
    char nombre[NOM];
    int edad, pasos=0;
    float altura, peso, zancada;
}PERFILES;

float zanc(float);
float IMC(float, float);
float dist(float, int);
float kcal(char, float, float, int);
int ficheroexiste(PERFILES*, int);
//float velmedia(float metros, float tiempo);

int main()
{
    //Arduino
    /*-----*/
    /*char a[] = { "\\.\COM5" }, *port_name = a;
    int b = 0;

    SerialPort arduino(port_name);

    Sleep(500);
    char ans[] = { "88888a" };
    arduino.writeSerialPort(ans, 15);

    //Conexion por serial con Arduino
    using System.IO.Ports;
```

```

namespace RadarSonarWPF.Class
{
    public class ComunicacionPuertoSerie
    {
        // Nueva instancia de la clase
        SerialPort serialPort = new SerialPort();

        public ComunicacionPuertoSerie()
        {
            // Asignamos las propiedades
            serialPort.BaudRate = 9600;
            serialPort.PortName = "COM5";

            // Creamos el evento
            serialPort.DataReceived += new
SerialDataReceivedEventHandler(SerialPort_DataReceived);

            // Controlamos que el puerto indicado esté operativo
            try
            {
                // Abrimos el puerto serie
                serialPort.Open();
            }
            catch
            {
            }
        }

        private void SerialPort_DataReceived(object sender,
SerialDataReceivedEventArgs e)
        {
            // Obtenemos el puerto serie que lanza el evento
            SerialPort currentSerialPort = (SerialPort)sender;

            // Leemos el dato recibido del puerto serie
            string inData = currentSerialPort.ReadLine();
        }
    }
}*/
/*-----*/

PERFILES p1, p2, p3;
PERFILES *perf1, *perf2, *perf3;

```

```

char opcgen;
int opcmenu, opczanc, pasostot, numperf, flag; //edad;
float indice, kcaloria, distancia; //altura, zancada, peso;
//char nombre[NOM]; //Solo hay 3 perfiles
//char nombre1[NOM], nombre2[NOM], nombre3[NOM];

do
{
printf("Bienvenido a la seccion interactiva de Gymnastiksko.\n");
printf("\nMENU\n");
printf("[1]-Entrar en un perfil existente\n[2]-Sobreescribir un perfil existente\n[3]-Crear
un nuevo perfil\n[4] - Salir\n\n");
scanf_s("%d", &opcmenu);

switch (opcmenu)
{
case 1: //Perfil existente
//Ficheros CON READ
int opcperf;
FILE *pperfil1;
printf("Que archivo quiere abrir? (1, 2 o 3)\n");
scanf_s("%d", &opcperf);

switch (opcperf)
{
case 1:
errno_t err11; //TOTALMENTE NECESARIO
err11 = fopen_s(&pperfil1, "1.txt", "r+"); //Solo lo queremos leer
el archivo

//Check-in del errno

if (err11)
{
printf("El archivo no se ha podido abrir.\n");
system("PAUSE");
}
else
{
printf("Archivo abierto correctamente.\n");

while (!feof(pperfil1))
fscanf_s(pperfil1, "%s %d %f %f %f %d",
p1.nombre, _countof(p1.nombre), &p1.edad, &p1.altura, &p1.peso, &p1.zancada,
&p1.pasos); //Para leer el archivo

```

```

printf("Nombre %s\n", p1.nombre);
printf("Edad %d\n", p1.edad);
printf("Altura %.2f\n", p1.altura);
printf("Peso %.2f\n", p1.peso);
printf("Zancada %.2f\n", p1.zancada);
printf("Numero de pasos dados según la informacion
dada por la zapatilla en su ultimo uso %d\n", p1.pasos);

```

opcgen = 'H'; //Tomamos como referencia los datos de un hombre para que pueda trabajar bajo una referencia

```

kcaloria = kcal(opcgen, p1.peso, p1.pasos, p1.edad);
printf("Las calorías quemadas, tal y como debe
aparecer en el dispositivo son %.2f\n", kcaloria);
distancia = dist(p1.zancada, p1.pasos);
printf("La distancia recorrida por el usuario de
GYMNASTIKSKO es %.2f m \n", distancia);
indice = IMC(p1.altura, p1.peso);
printf("Segun los datos introducidos por el usuario su
IMC es %.2f\n", indice);

```

```

while (!feof(pperfil1))
    fscanf_s(pperfil1, "%s %d %f %f %f %d %f %f
%f", p1.nombre, _countof(p1.nombre), &p1.edad, &p1.altura, &p1.peso, &p1.zancada,
&p1.pasos, &kcaloria, &distancia, &indice);

```

```

fprintf(pperfil1, "\n");
fprintf(pperfil1, "%.2f\n", kcaloria);
fprintf(pperfil1, "%.2f\n", distancia);
fprintf(pperfil1, "%.2f\n", indice);

```

```

//Cerramos todos los archivos
fclose(pperfil1);

```

```

}
break;

```

case 2:

```

errno_t err12; //TOTALMENTE NECESARIO

```

```

err12 = fopen_s(&pperfil1, "2.txt", "r+"); //Solo lo queremos leer

```

el archivo

```

//Check-in del errno

```

```

if (err12)
{
    printf("El archivo no se ha podido abrir.\n");
    system("PAUSE");
}

```

```

    }
    else
    {
        printf("Archivo abierto correctamente.\n");

        while (!feof(pperfil1))
            fscanf_s(pperfil1, "%s %d %f %f %f %d",
p2.nombre, _countof(p2.nombre), &p2.edad, &p2.altura, &p2.peso, &p2.zancada,
&p2.pasos); //Para leer el archivo

        printf("Nombre %s\n", p2.nombre);
        printf("Edad %d\n", p2.edad);
        printf("Altura %.2f\n", p2.altura);
        printf("Peso %.2f\n", p2.peso);
        printf("Zancada %.2f\n", p2.zancada);
        printf("Numero de pasos dados según la informacion
dada por la zapatilla en su ultimo uso %d\n", p2.pasos);

        opcgen = 'H'; //Tomamos como referencia los datos de
un hombre para que pueda trabajar bajo una referencia

        kcaloria = kcal(opcgen, p2.peso, p2.pasos, p2.edad);
        printf("Las calorías quemadas, tal y como debe
aparecer en el dispositivo son %.2f\n", kcaloria);
        distancia = dist(p2.zancada, p2.pasos);
        printf("La distancia recorrida por el usuario de
GYMNASTIKSKO es %.2f m \n", distancia);
        indice = IMC(p2.altura, p2.peso);
        printf("Segun los datos introducidos por el usuario su
IMC es %.2f\n", indice);

        while (!feof(pperfil1))
            fscanf_s(pperfil1, "%s %d %f %f %f %d %f %f
%f", p2.nombre, _countof(p2.nombre), &p2.edad, &p2.altura, &p2.peso, &p2.zancada,
&p2.pasos, &kcaloria, &distancia, &indice);

        fprintf(pperfil1, "\n");
        fprintf(pperfil1, "%.2f\n", kcaloria);
        fprintf(pperfil1, "%.2f\n", distancia);
        fprintf(pperfil1, "%.2f\n", indice);

        //Cerramos todos los archivos
        fclose(pperfil1);
    }
    break;
case 3:

```

```

        errno_t err13; //TOTALMENTE NECESARIO
        err13 = fopen_s(&pperfil1, "3.txt", "r+"); //Solo lo queremos leer
el archivo

        //Check-in del errno

        if (err13)
        {
            printf("El archivo no se ha podido abrir.\n");
            system("PAUSE");
        }
        else
        {
            printf("Archivo abierto correctamente.\n");

            while (!feof(pperfil1))
                fscanf_s(pperfil1, "%s %d %f %f %f %d",
p3.nombre, _countof(p3.nombre), &p3.edad, &p3.altura, &p3.peso, &p3.zancada,
&p3.pasos); //Para leer el archivo

            printf("Nombre %s\n", p3.nombre);
            printf("Edad %d\n", p3.edad);
            printf("Altura %.2f\n", p3.altura);
            printf("Peso %.2f\n", p3.peso);
            printf("Zancada %.2f\n", p3.zancada);
            printf("Numero de pasos dados según la informacion
dada por la zapatilla en su ultimo uso %d\n", p3.pasos);

            opcgen = 'H'; //Tomamos como referencia los datos de
un hombre para que pueda trabajar bajo una referencia

            kcaloria = kcal(opcgen, p3.peso, p3.pasos, p3.edad);
            printf("Las calorías quemadas, tal y como debe
aparecer en el dispositivo son %.2f\n", kcaloria);
            distancia = dist(p3.zancada, p3.pasos);
            printf("La distancia recorrida por el usuario de
GYMNASTIKSKO es %.2f m \n", distancia);
            indice = IMC(p3.altura, p3.peso);
            printf("Segun los datos introducidos por el usuario su
IMC es %.2f\n", indice);

            while (!feof(pperfil1))
                fscanf_s(pperfil1, "%s %d %f %f %f %d %f %f
%f", p3.nombre, _countof(p3.nombre), &p3.edad, &p3.altura, &p3.peso, &p3.zancada,
&p3.pasos, &kcaloria, &distancia, &indice);

```



```

        fprintf(pperfil1, "\n");
        fprintf(pperfil1, "%.2f\n", kcaloria);
        fprintf(pperfil1, "%.2f\n", distancia);
        fprintf(pperfil1, "%.2f\n", indice);

        //Cerramos todos los archivos
        fclose(pperfil1);
    }
    break;
default:
    printf("Archivo no existente.\n");
}
break;
case 2: //Sobreescribir un perfil ya existente
    //Ficheros CON WRITE
    char perfil[NOM];
    FILE *pperfil2;
    printf("Introduzca el nombre del archivo:\n");
    scanf_s("%d", &opcperf);

    switch (opcperf)
    {
    case 1:
        errno_t err21; //TOTALMENTE NECESARIO
        err21 = fopen_s(&pperfil2, "1.txt", "w+");

        if (err21)
        {
            printf("El archivo no se ha podido abrir.\n");
            system("PAUSE");
        }
        else
        {
            printf("Archivo abierto correctamente.\n");

            printf("Introduzca los siguientes parametros:\n");
            printf("Nombre:\n");
            getchar();
            gets_s(p1.nombre, NOM);
            printf("Edad:\n");
            scanf_s("%d", &p1.edad);
            printf("Altura:\n");
            scanf_s("%f", &p1.altura);
            printf("Peso:\n");
            scanf_s("%f", &p1.peso);
            printf("Zancada:\n");

```

```

scanf_s("%f", &p1.zancada);
printf("Numero de pasos dados según la informacion
dada por la zapatilla en su ultimo uso:\n");
scanf_s("%d", &p1.pasos);
getchar();
printf("Es usted: \n[H] - Hombre\n[M] - Mujer\n");
scanf_s("%c", &opcggen);

while (!feof(pperfil2))
    fscanf_s(pperfil2, "%s %d %f %f %f %d",
p1.nombre, _countof(p1.nombre), &p1.edad, &p1.altura, &p1.peso, &p1.zancada,
&p1.pasos);

fprintf(pperfil2, "%s\n", p1.nombre);
fprintf(pperfil2, "%d\n", p1.edad);
fprintf(pperfil2, "%.2f\n", p1.altura);
fprintf(pperfil2, "%.2f\n", p1.peso);
fprintf(pperfil2, "%.2f\n", p1.zancada);

kcaloria = kcal(opcggen, p1.peso, p1.pasos, p1.edad);
printf("Las calorías quemadas, tal y como debe
aparecer en el dispositivo son %.2f\n", kcaloria);
distancia = dist(p1.zancada, p1.pasos);
printf("La distancia recorrida por el usuario de
GYMNASTIKSKO es %.2f m \n", distancia);
indice = IMC(p1.altura, p1.peso);
printf("Según los datos introducidos por el usuario su
IMC es %.2f\n", indice);

while (!feof(pperfil2))
    fscanf_s(pperfil2, "%s %d %f %f %f %d %f %f
%f", p1.nombre, _countof(p1.nombre), &p1.edad, &p1.altura, &p1.peso, &p1.zancada,
&p1.pasos, &kcaloria, &distancia, &indice);

fprintf(pperfil2, "\n");
fprintf(pperfil2, "%.2f\n", kcaloria);
fprintf(pperfil2, "%.2f\n", distancia);
fprintf(pperfil2, "%.2f\n", indice);

//Cerramos todos los archivos
fclose(pperfil2);
}
break;
case 2:
    errno_t err22; //TOTALMENTE NECESARIO
    err22 = fopen_s(&pperfil2, "2.txt", "w+");

```

```

if (err22)
{
    printf("El archivo no se ha podido abrir.\n");
    system("PAUSE");
}
else
{
    printf("Archivo abierto correctamente.\n");

    printf("Introduzca los siguientes parametros:\n");
    printf("Nombre:\n");
    getchar();
    gets_s(p1.nombre, NOM);
    printf("Edad:\n");
    scanf_s("%d", &p2.edad);
    printf("Altura:\n");
    scanf_s("%f", &p2.altura);
    printf("Peso:\n");
    scanf_s("%f", &p2.peso);
    printf("Zancada:\n");
    scanf_s("%f", &p2.zancada);
    printf("Numero de pasos dados según la informacion
dada por la zapatilla en su ultimo uso:\n");
    scanf_s("%d", &p2.pasos);
    getchar();
    printf("Es usted: \n[H] - Hombre\n[M] - Mujer\n");
    scanf_s("%c", &opcggen);

    while (!feof(pperfil2))
        fscanf_s(pperfil2, "%s %d %f %f %f %d",
p2.nombre, _countof(p2.nombre), &p2.edad, &p2.altura, &p2.peso, &p2.zancada,
&p2.pasos);

    fprintf(pperfil2, "%s\n", p2.nombre);
    fprintf(pperfil2, "%d\n", p2.edad);
    fprintf(pperfil2, "%.2f\n", p2.altura);
    fprintf(pperfil2, "%.2f\n", p2.peso);
    fprintf(pperfil2, "%.2f\n", p2.zancada);

    kcaloria = kcal(opcggen, p2.peso, p2.pasos, p2.edad);
    printf("Las calorías quemadas, tal y como debe
aparecer en el dispositivo son %.2f\n", kcaloria);
    distancia = dist(p2.zancada, p2.pasos);
    printf("La distancia recorrida por el usuario de
GYMNASTIKSKO es %.2f m \n", distancia);

```

```

        indice = IMC(p2.altura, p2.peso);
        printf("Segun los datos introducidos por el usuario su
IMC es %.2f\n", indice);

        while (!feof(pperfil2))
            fscanf_s(pperfil2, "%s %d %f %f %d %f %f
%f", p2.nombre, _countof(p2.nombre), &p2.edad, &p2.altura, &p2.peso, &p2.zancada,
&p2.pasos, &kcaloria, &distancia, &indice);

        fprintf(pperfil2, "\n");
        fprintf(pperfil2, "%.2f\n", kcaloria);
        fprintf(pperfil2, "%.2f\n", distancia);
        fprintf(pperfil2, "%.2f\n", indice);

        //Cerramos todos los archivos
        fclose(pperfil2);
    }
    break;
case 3:
    errno_t err23; //TOTALMENTE NECESARIO
    err23 = fopen_s(&pperfil2, "3.txt", "w+");

    if (err23)
    {
        printf("El archivo no se ha podido abrir.\n");
        system("PAUSE");
    }
    else
    {
        printf("Archivo abierto correctamente.\n");

        printf("Introduzca los siguientes parametros:\n");
        printf("Nombre:\n");
        getchar();
        gets_s(p3.nombre, NOM);
        printf("Edad:\n");
        scanf_s("%d", &p3.edad);
        printf("Altura:\n");
        scanf_s("%f", &p3.altura);
        printf("Peso:\n");
        scanf_s("%f", &p3.peso);
        printf("Zancada:\n");
        scanf_s("%f", &p3.zancada);
        printf("Numero de pasos dados según la informacion
dada por la zapatilla en su ultimo uso:\n");
        scanf_s("%d", &p3.pasos);
    }
}

```

```

        getchar();
        printf("Es usted: \n[H] - Hombre\n[M] - Mujer\n");
        scanf_s("%c", &opcggen);

        while (!feof(pperfil2))
            fscanf_s(pperfil2, "%s %d %f %f %f %d",
p3.nombre, _countof(p3.nombre), &p3.edad, &p3.altura, &p3.peso, &p3.zancada,
&p3.pasos);

        fprintf(pperfil2, "%s\n", p3.nombre);
        fprintf(pperfil2, "%d\n", p3.edad);
        fprintf(pperfil2, "%.2f\n", p3.altura);
        fprintf(pperfil2, "%.2f\n", p3.peso);
        fprintf(pperfil2, "%.2f\n", p3.zancada);

        kcaloria = kcal(opcggen, p3.peso, p3.pasos, p3.edad);
        printf("Las calorías quemadas, tal y como debe
aparecer en el dispositivo son %.2f\n", kcaloria);
        distancia = dist(p3.zancada, p3.pasos);
        printf("La distancia recorrida por el usuario de
GYMNASTIKSKO es %.2f m \n", distancia);
        indice = IMC(p3.altura, p3.peso);
        printf("Según los datos introducidos por el usuario su
IMC es %.2f\n", indice);

        while (!feof(pperfil2))
            fscanf_s(pperfil2, "%s %d %f %f %f %d %f %f
%f", p3.nombre, _countof(p3.nombre), &p3.edad, &p3.altura, &p3.peso, &p3.zancada,
&p3.pasos, &kcaloria, &distancia, &indice);

        fprintf(pperfil2, "\n");
        fprintf(pperfil2, "%.2f\n", kcaloria);
        fprintf(pperfil2, "%.2f\n", distancia);
        fprintf(pperfil2, "%.2f\n", indice);

        //Cerramos todos los archivos
        fclose(pperfil2);
    }
    break;
default:
    printf("Archivo no existente.\n");
}
break;
case 3: //Crear un perfil nuevo
    //Comprobación: está escrito el fichero?

```

```

//Inicializacion de los punteros
perf1 = &p1;
perf2 = &p2;
perf3 = &p3;

do
{
    printf("Introduzca el nombre del archivo donde desee guardar
la informacion.\n");

    scanf_s("%d", &opcperf);

    if (opcperf == 1)
    {
        flag = ficheroexiste(perf1, opcperf);

        FILE *pperfil41;
        pperfil41 = 0;

        printf("Introduzca los siguientes parametros:\n");
        printf("Nombre:\n");
        getchar();
        gets_s(p1.nombre, NOM);
        printf("Edad:\n");
        scanf_s("%d", &p1.edad);
        printf("Altura:\n");
        scanf_s("%f", &p1.altura);
        printf("Peso:\n");
        scanf_s("%f", &p1.peso);
        printf("Zancada:\n");
        scanf_s("%f", &p1.zancada);
        printf("Numero de pasos dados segun la informacion
dada por la zapatilla en su ultimo uso:\n");
        scanf_s("%d", &p1.pasos);
        getchar();
        printf("Es usted: \n[H] - Hombre\n[M] - Mujer\n");
        scanf_s("%c", &opcgen);

        while (!feof(pperfil41))
            fscanf_s(pperfil41, "%s %d %f %f %f %d",
p1.nombre, _countof(p1.nombre), &p1.edad, &p1.altura, &p1.peso, &p1.zancada,
&p1.pasos);

        fprintf(pperfil41, "%s\n", p1.nombre);
        fprintf(pperfil41, "%d\n", p1.edad);
        fprintf(pperfil41, "%.2f\n", p1.altura);
        fprintf(pperfil41, "%.2f\n", p1.peso);
    }
}

```

```

        fprintf(pperfil41, "%.2f\n", p1.zancada);

        kcaloria = kcal(opcgen, p1.peso, p1.pasos, p1.edad);
        printf("Las calorías quemadas, tal y como debe
aparecer en el dispositivo son %.2f\n", kcaloria);
        distancia = dist(p1.zancada, p1.pasos);
        printf("La distancia recorrida por el usuario de
GYMNASTIKSKO es %.2f m \n", distancia);
        indice = IMC(p1.altura, p1.peso);
        printf("Según los datos introducidos por el usuario su
IMC es %.2f\n", indice);

        while (!feof(pperfil41))
            fscanf_s(pperfil41, "%s %d %f %f %f %d %f %f
%f", p1.nombre, _countof(p1.nombre), &p1.edad, &p1.altura, &p1.peso, &p1.zancada,
&p1.pasos, &kcaloria, &distancia, &indice);

        fprintf(pperfil41, "\n");
        fprintf(pperfil41, "%.2f\n", kcaloria);
        fprintf(pperfil41, "%.2f\n", distancia);
        fprintf(pperfil41, "%.2f\n", indice);

        //Cerramos todos los archivos
        fclose(pperfil41);
    }
    else
    {
        if (opcperf == 2)
        {
            flag = ficheroexiste(perf2, opcperf);

            FILE *pperfil42;
            pperfil42 = 0;

            printf("Introduzca los siguientes parámetros:\n");
            printf("Nombre:\n");
            getchar();
            gets_s(p2.nombre, NOM);
            printf("Edad:\n");
            scanf_s("%d", &p2.edad);
            printf("Altura:\n");
            scanf_s("%f", &p2.altura);
            printf("Peso:\n");
            scanf_s("%f", &p2.peso);
            printf("Zancada:\n");
            scanf_s("%f", &p2.zancada);

```

```

        printf("Numero de pasos dados según la
informacion dada por la zapatilla en su ultimo uso:\n");
        scanf_s("%d", &p2.pasos);
        getchar();
        printf("Es usted: \n[H] - Hombre\n[M] - Mujer\n");
        scanf_s("%c", &opcgen);

        while (!feof(pperfil42))
            fscanf_s(pperfil42, "%s %d %f %f %f %f
%d", p2.nombre, _countof(p2.nombre), &p2.edad, &p2.altura, &p2.peso, &p2.zancada,
&p2.pasos);

        fprintf(pperfil42, "%s\n", p2.nombre);
        fprintf(pperfil42, "%d\n", p2.edad);
        fprintf(pperfil42, "%.2f\n", p2.altura);
        fprintf(pperfil42, "%.2f\n", p2.peso);
        fprintf(pperfil42, "%.2f\n", p2.zancada);

        kcaloria = kcal(opcgen, p2.peso, p2.pasos,
p2.edad);

        printf("Las calorías quemadas, tal y como debe
aparecer en el dispositivo son %.2f\n", kcaloria);
        distancia = dist(p2.zancada, p2.pasos);
        printf("La distancia recorrida por el usuario de
GYMNASTIKSKO es %.2f m \n", distancia);
        indice = IMC(p2.altura, p2.peso);
        printf("Segun los datos introducidos por el
usuario su IMC es %.2f\n", indice);

        while (!feof(pperfil42))
            fscanf_s(pperfil42, "%s %d %f %f %f %f %d
%f %f %f", p2.nombre, _countof(p2.nombre), &p2.edad, &p2.altura, &p2.peso, &p2.zancada,
&p2.pasos, &kcaloria, &distancia, &indice);

        fprintf(pperfil42, "\n");
        fprintf(pperfil42, "%.2f\n", kcaloria);
        fprintf(pperfil42, "%.2f\n", distancia);
        fprintf(pperfil42, "%.2f\n", indice);

        //Cerramos todos los archivos
        fclose(pperfil42);
    }
    else
    {
        flag = ficheroexiste(perf3, opcperf);

```



```

FILE *pperfil43;
pperfil43 = 0;

printf("Introduzca los siguientes parametros:\n");
printf("Nombre:\n");
getchar();
gets_s(p3.nombre, NOM);
printf("Edad:\n");
scanf_s("%d", &p3.edad);
printf("Altura:\n");
scanf_s("%f", &p3.altura);
printf("Peso:\n");
scanf_s("%f", &p3.peso);
printf("Zancada:\n");
scanf_s("%f", &p3.zancada);
printf("Numero de pasos dados según la
informacion dada por la zapatilla en su ultimo uso:\n");
scanf_s("%d", &p3.pasos);
getchar();
printf("Es usted: \n[H] - Hombre\n[M] - Mujer\n");
scanf_s("%c", &opcgen);

while (!feof(pperfil43))
    fscanf_s(pperfil43, "%s %d %f %f %f
%d", p3.nombre, _countof(p3.nombre), &p3.edad, &p3.altura, &p3.peso, &p3.zancada,
&p3.pasos);

fprintf(pperfil43, "%s\n", p3.nombre);
fprintf(pperfil43, "%d\n", p3.edad);
fprintf(pperfil43, "%.2f\n", p3.altura);
fprintf(pperfil43, "%.2f\n", p3.peso);
fprintf(pperfil43, "%.2f\n", p3.zancada);

kcaloria = kcal(opcgen, p3.peso, p3.pasos,
p3.edad);

printf("Las calorías quemadas, tal y como debe
aparecer en el dispositivo son %.2f\n", kcaloria);
distancia = dist(p3.zancada, p3.pasos);
printf("La distancia recorrida por el usuario de
GYMNASTIKSKO es %.2f m \n", distancia);
indice = IMC(p3.altura, p3.peso);
printf("Segun los datos introducidos por el
usuario su IMC es %.2f\n", indice);

while (!feof(pperfil43))

```

```

fscanf_s(ppperfil43, "%s %d %f %f %f %d
%f %f %f", p3.nombre, _countof(p3.nombre), &p3.edad, &p3.altura, &p3.peso, &p3.zancada,
&p3.pasos, &kcaloria, &distancia, &indice);

```

```

fprintf(ppperfil43, "\n");
fprintf(ppperfil43, "%.2f\n", kcaloria);
fprintf(ppperfil43, "%.2f\n", distancia);
fprintf(ppperfil43, "%.2f\n", indice);

```

```

//Cerramos todos los archivos
fclose(ppperfil43);

```

```

    }
}

```

```

printf("Archivo no disponible. Introduzca otro archivo");

```

```

}
while (flag == 0);

```

```

break;

```

```

//Falta meter todo esto nuevo en un fichero en funcion de WRITE
/*int opcperf3;
char perfil31[NOM];
FILE *ppperfil31;*/
break;

```

```

case 4:
printf("Adios.\n");
break;

```

```

default:
printf("La opcion requerida no se encuentra disponible.\n");

```

```

}

```

```

} while (opcmenu != 4);

```

```

printf("\n");
system("PAUSE");

```

```

}

```

```

float zanc(float alt) //Falta la llamada a la funcion arriba en el main()

```

```

{

```

```

    char sexo;
    float altura, zancpredeth, zancpredetm, zancpredetn;
    printf("Indique su sexo : \n[M] - Masculino\n[F] - Femenino\n[N] - Prefiero no
indicarlo\n");
    getchar();
    scanf_s("%c", &sexo);

```

```

switch (sexo)
{
case 'M':
    zancpredeth = alt * 0.415; //Andando
    break;
case 'F':
    zancpredetm = alt * 0.413; //Andando
    break;
case 'N':
    zancpredetn = alt * 0.414; //Andando
    break;
default:
    printf("La opcion no se encuentra disponible. Discupe las molestias.\n");
}

if (sexo == 'M')
    return zancpredeth;

if (sexo == 'F')
    return zancpredetm;

if (sexo == 'N')
    return zancpredetn;
}

float IMC(float m, float kg)
{
    return(kg / (m*m));
}

float dist(float zanc, int pasos)
{
    return zanc * pasos;
}

float kcal(char gen, float kg, float m, int anho) //Falta aun su llamada arriba en el main()
{
    float resul;

    switch (gen)
    {
    case 'H':
        resul = 13.75*kg + 5 * m - 6.76*anho + 66;
        return resul;
        break;
    case 'M':

```

```

        resul = 9.65*kg + 1.85 * m - 4.68*anho + 655;
        return resul;
        break;
    default:
        printf("La opcion no se encuentra disponible. Discupe las molestias.\n");
    }
}

int ficheroexiste(PERFILES *p, int archivo)
{
    int idx, flag;

    switch (archivo)
    {
    case 1:
        FILE *pperfil31;
        errno_t err31; //TOTALMENTE NECESARIO
        err31 = fopen_s(&pperfil31, "1.txt", "r");

        while (!feof(pperfil31))
            fscanf_s(pperfil31, "%s", &(p->nombre), _countof(p->nombre)); //Para
leer tan solo el nombre escrito en el archivo que es lo unico que nos interesa

        if (p->nombre[0] >= 0 && p->nombre[0] <= 255)
        {
            printf("El archivo esta escrito.\n");
            idx = 1;
        }
        else
        {
            printf("El archivo no esta escrito.\n");
            idx = 0;
        }
        break;
    case 2:
        FILE *pperfil32;
        errno_t err32; //TOTALMENTE NECESARIO
        err31 = fopen_s(&pperfil32, "2.txt", "r");

        while (!feof(pperfil32))
            fscanf_s(pperfil32, "%s", &(p->nombre), _countof(p->nombre)); //Para
leer tan solo el nombre escrito en el archivo que es lo unico que nos interesa

        if (p->nombre[0] >= 0 && p->nombre[0] <= 255)
        {
            printf("El archivo esta escrito.\n");

```

```

        idx = 1;
    }
    else
    {
        printf("El archivo no esta escrito.\n");
        idx = 0;
    }
    break;
case 3:
    FILE *pperfil33;
    errno_t err33; //TOTALMENTE NECESARIO
    err31 = fopen_s(&pperfil33, "3.txt", "r");

    while (!feof(pperfil33))
        fscanf_s(pperfil33, "%s", &(p->nombre), _countof(p->nombre)); //Para
leer tan solo el nombre escrito en el archivo que es lo unico que nos interesa

    if (p->nombre[0] >= 0 && p->nombre[0] <= 255)
    {
        printf("El archivo esta escrito.\n");
        idx = 1;
    }
    else
    {
        printf("El archivo no esta escrito.\n");
        idx = 0;
    }
    break;
default:
    printf("Nombre del archivo no valido.");
}

return idx;
}

```

## CÓDIGO EN ARDUINO

```
//librerias/variables de lcd
#include <FastIO.h>
#include <I2CIO.h>
#include <LCD.h>
#include <LiquidCrystal.h>
#include <LiquidCrystal_I2C.h>
#include <LiquidCrystal_SR.h>
#include <LiquidCrystal_SR2W.h>
#include <LiquidCrystal_SR3W.h>
#include <Wire.h>

LiquidCrystal_I2C lcd(0x27,2,1,0,4,5,6,7,3,POSITIVE); //declaracion de direccion de lcd
byte exclamacion[8] = { //byte de 8x5 declaramos caracter especial dibujandolo con 0 y 1
  0b00100,
  0b00000,
  0b00000,
  0b00100,
  0b00100,
  0b00100,
  0b00100,
  0b00000
};
//variables de sensor FRS
int AnalogPin = 0; // Sensor conectado a Analog 0
int ResRead; // La Lectura de la Resistencia por División de Tensión
int contador=0;
boolean piso=false, levanto=false;
//boton
int boton=0;
const int pulsadorPin=2;
int valorPulsador=0;
boolean pulsado= 0, soltado=0;
//funciones
float calorias(float, float,int);
float distancia(float,int);
float fdist=0, fcal=0;
float zancada=0.65;
float peso=73.0, altura=1.81, edad=23;

void setup()
{
  Serial.begin(9600);
  pinMode(pulsadorPin, INPUT);
```

```

lcd.begin(16,2);//Definimos 16 columnas y 2 filas del lcd
lcd.backlight();//Encendemos luz negra
lcd.backlight();//Apagamos luz negra--- esto indica que empieza el programa
lcd.createChar(0, exclamacion);
lcd.setCursor(0,0);//POSICION CURSOR COLUMNA, FILA
lcd.print(" GYMNASTIKSKO");
delay(4000);
lcd.clear();//LIMPIA LA PANTALLA
lcd.setCursor(0,0);
lcd.print("Vamos a contar");
lcd.setCursor(0,1);
lcd.print("tus pasos");
delay(4000);
lcd.clear();
lcd.setCursor(3,0);
lcd.write((byte)0);
lcd.print("EMPECEMOS! ");
delay(2000);
lcd.clear();
}
void loop()
{
  do{
    valorPulsador=digitalRead(pulsadorPin);
    if (valorPulsador== 1)
    {
      pulsado=true;
      while(valorPulsador)
      {
        valorPulsador =digitalRead(pulsadorPin);
        Serial.print("Pulso=");
        Serial.println(pulsado);
      }
      Serial.println(valorPulsador);
      if(valorPulsador==0)
      {
        soltado=true;
      }
    }
    if(pulsado==true && soltado==true)
    {
      boton++;
    }
  }do{
    //FUNCIONES PARA CALORIAS Y KM
    fdist= distancia(zancada, contador);
  }
}

```

```

        //CALORIAS
        fcal=calorias(peso,fdist,edad);
        //CAMBIO DE MENSAJITOS //PULSADOR
        valorPulsador=digitalRead(pulsadorPin);
        if (valorPulsador== true)
        {
            pulsado=true;
            while(valorPulsador)
            {
                valorPulsador =digitalRead(pulsadorPin);
                Serial.print("Pulso=");
                Serial.println(pulsado);
            }
            Serial.println(valorPulsador);
            if(valorPulsador==0)
            {
                soltado=true;
            }
        }
        if(pulsado==true && soltado==true)
        {
            boton++;
        }
//SENSOR CUENTA PASOS
        valorPulsador=digitalRead(pulsadorPin);
        ResRead = analogRead(AnalogPin); // La Resistencia es igual a la lectura del sensor
        (Analog 0)
        Serial.print("Lectura Analogica = ");
        Serial.println(ResRead);
        if (ResRead>=800)
        {
            piso=true;
            while(ResRead>=800)
            {
                ResRead = analogRead(AnalogPin);
                Serial.print("Piso=");
                Serial.println(piso);
            }
            if(ResRead<800 && ResRead>=200)
            {
                while(ResRead<800 && ResRead>=200)
                {
                    ResRead = analogRead(AnalogPin);
                    Serial.println("Sigue levantando");
                }
            }
        }

```



```

    if(ResRead<200)
    {
        levanto=true;
        Serial.print("Levanto=");
        Serial.println(levanto);
    }
}
if(levanto==true && piso==true)
{
    contador++;
    Serial.println("Contador=");
    Serial.print(contador);
}

valorPulsador=digitalRead(pulsadorPin);
lcd.setCursor(0,0);
lcd.print("Numero de pasos:");
lcd.setCursor(0,1);
lcd.print(contador);
piso=false;
levanto=false;
pulsado=false;
soltado=false;
b++;
}while(boton==0);

lcd.clear();
do
{
    fdist= distancia(zancada, contador);
    //CALORIAS
    fcal=calorias(peso,fdist,edad);
    //CAMBIO DE MENSAJITOS //PULSADOR
    valorPulsador=digitalRead(pulsadorPin);
    if (valorPulsador== 1)
    {
        pulsado=true;
        while(valorPulsador)
        {
            valorPulsador =digitalRead(pulsadorPin);
            Serial.print("Pulso=");
            Serial.println(pulsado);
        }
        Serial.println(valorPulsador);
        if(valorPulsador==0)
        {
            soltado=true;

```

```

    }
}
if(pulsado==true && soltado==true)
{
    boton++;
}

//SENSOR CUENTA PASOS
valorPulsador=digitalRead(pulsadorPin);
ResRead = analogRead(AnalogPin); // La Resistencia es igual a la lectura del sensor
(Analog 0)
Serial.print("Lectura Analogica = ");
Serial.println(ResRead);
if (ResRead>=800)
{
    piso=true;
    while(ResRead>=800)
    {
        ResRead = analogRead(AnalogPin);
        Serial.print("Piso=");
        Serial.println(piso);
    }
    if(ResRead<800 && ResRead>=200)
    {
        while(ResRead<800 && ResRead>=200)
        {
            ResRead = analogRead(AnalogPin);
            Serial.println("Sigue levantando");
        }
    }
    if(ResRead<200)
    {
        levanto=true;
        Serial.print("Levanto=");
        Serial.println(levanto);
    }
}
if(levanto==true && piso==true)
{
    contador++;
    Serial.println("Contador=");
    Serial.print(contador);
}
lcd.setCursor(0,0);
lcd.print("Kcal:");
lcd.setCursor(0,1);

```

```

        lcd.print(fcal);
        lcd.setCursor(8,0);
        lcd.print("Dist:");
        lcd.setCursor(8,1);
        lcd.print(fdist);
        piso=false;
        levanto=false;
        pulsado=false;
        soltado=false;
        b++;
    }while(boton==1);
    lcd.clear();
    boton=0;

}while(boton<=2);
lcd.clear();
lcd.setCursor(0,0);
lcd.print("Guardo datos");
delay(3000);
boton=0;
}
//funciones
float distancia(float zancada, int contador)
{
    float fdista=0;
    fdista=zancada*contador;
    return(fdista);
}
float calorías(float peso, float fdist, int edad)
{
    float fcal=0;
    fcal=(13.75*peso+5*fdist-6.76*edad+66)*0.001;
    //si es mujer: fcal=(9.65*peso+1.85*fdist-4.68*edad+655)*0.001;
    return(fcal);
}

```

## MEMORIA

“*Gymnastiksko*” es, básicamente, la traducción del sueco de “Zapatilla deportiva”, base fundamental de nuestro proyecto.

Nuestra zapatilla tendrá como finalidad alcanzar las funciones típicas de las actuales *smart brands* como las de *Xiaomi* o *Apple*, entre otras. Esto es: que sea capaz de contabilizar el número de pasos dados por el usuario y, a partir de ellos, calcular la distancia recorrida y las calorías quemadas.

Además de esas funciones que son las principales en Arduino, el programa en C será capaz de calcular el IMC y otros datos fundamentales para calcular las ya mencionadas kcal (por ejemplo).

Este código implementa el uso de estructuras para tratar los perfiles de los usuarios cuya información será posteriormente guardada en ficheros, base fundamental de nuestro proyecto.

Como tal, el programa permite al usuario elegir, desde la sección interactiva y mediante un menú, varias opciones que son:

- Entrar en un perfil existente
- Editar un perfil existente
- Crear un nuevo perfil

Teniendo en cuenta que, como máximo, se podrán tener tres únicos perfiles.

El código de Arduino está dividido entre los tres elementos más relevantes de la parte “física” del proyecto:

- Sensor resistivo de fuerza
- Pantalla LCD
- Pulsador