

# UNIVERSIDAD PRIVADA DE TACNA

## ESCUELA PROFESIONAL INGENIERÍA DE SISTEMAS



### Lab. Ficheros y Árboles

#### ESTUDIANTE:

- |                                     |      |
|-------------------------------------|------|
| - LUDEÑA BEDOYA, MILAGROS SHADU     | EPIS |
| - SEBASTIAN ALEJANDRO, CORTEZ APAZA | EPIS |

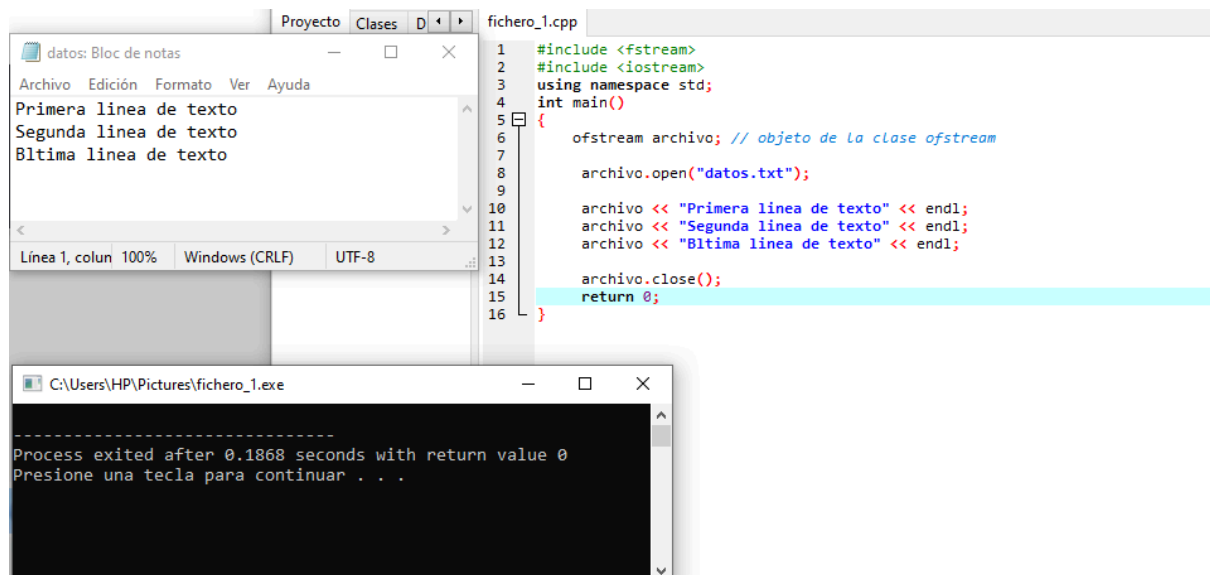
#### CURSO: ESTRUCTURA DE DATOS

#### DOCENTE: HAYDEE RAQUEL SISA YATAO

TACNA, 27 de noviembre de 2024

## Ejercicios - Ficheros:

1º Crear un fichero de texto llamado "datos.txt" y escribir en él tres líneas de texto:



The screenshot shows a C++ IDE with a file named 'fichero\_1.cpp'. The code uses `ofstream` to create and write to a file named 'datos.txt'. A separate window shows the content of 'datos.txt' with three lines of text. A console window shows the program's execution output.

```
1 #include <fstream>
2 #include <iostream>
3 using namespace std;
4 int main()
5 {
6     ofstream archivo; // objeto de la clase ofstream
7     archivo.open("datos.txt");
8
9     archivo << "Primera línea de texto" << endl;
10    archivo << "Segunda línea de texto" << endl;
11    archivo << "Última línea de texto" << endl;
12
13    archivo.close();
14    return 0;
15 }
```

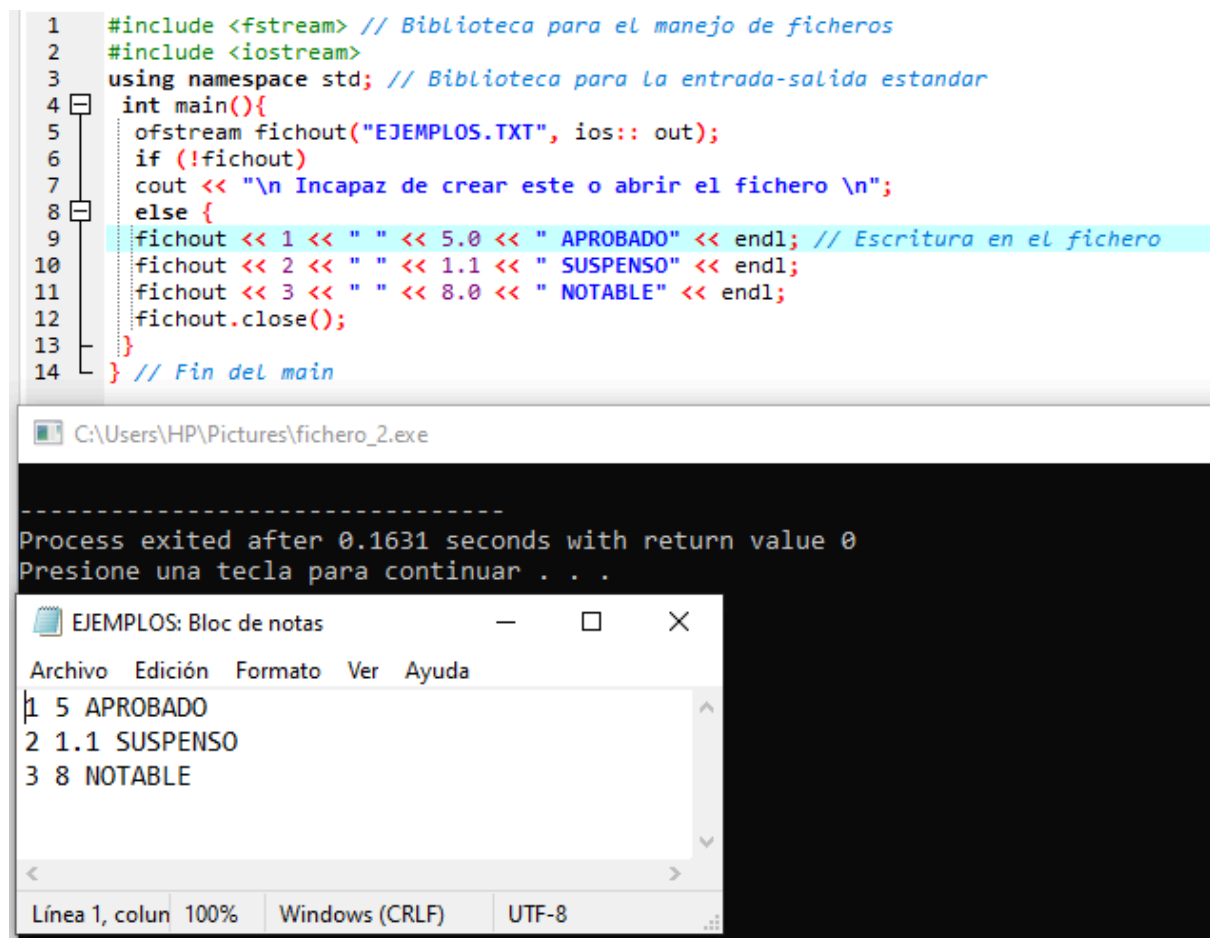
datos: Bloc de notas

Primera línea de texto  
Segunda línea de texto  
Última línea de texto

C:\Users\HP\Pictures\fichero\_1.exe

-----  
Process exited after 0.1868 seconds with return value 0  
Presione una tecla para continuar . . .

2º Crear un fichero de texto llamado "EJEMPLOS.TXT" y escribir en él información relacionada con el desempeño de varios estudiantes:



The screenshot shows a C++ IDE with a file named 'fichero\_2.cpp'. The code uses `ofstream` to create and write to a file named 'EJEMPLOS.TXT'. A separate window shows the content of 'EJEMPLOS.TXT' with three lines of student performance data. A console window shows the program's execution output.

```
1 #include <fstream> // Biblioteca para el manejo de ficheros
2 #include <iostream>
3 using namespace std; // Biblioteca para la entrada-salida estandar
4 int main(){
5     ofstream fichout("EJEMPLOS.TXT", ios:: out);
6     if (!fichout)
7         cout << "\n Incapaz de crear este o abrir el fichero \n";
8     else {
9         fichout << 1 << " " << 5.0 << " APROBADO" << endl; // Escritura en el fichero
10        fichout << 2 << " " << 1.1 << " SUSPENSO" << endl;
11        fichout << 3 << " " << 8.0 << " NOTABLE" << endl;
12        fichout.close();
13    }
14 } // Fin del main
```

C:\Users\HP\Pictures\fichero\_2.exe

-----  
Process exited after 0.1631 seconds with return value 0  
Presione una tecla para continuar . . .

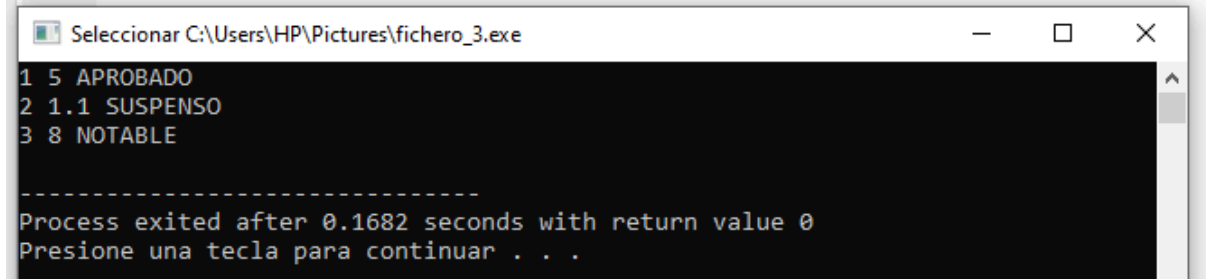
EJEMPLOS: Bloc de notas

1 5 APROBADO  
2 1.1 SUSPENSO  
3 8 NOTABLE

Línea 1, colun 100% Windows (CRLF) UTF-8

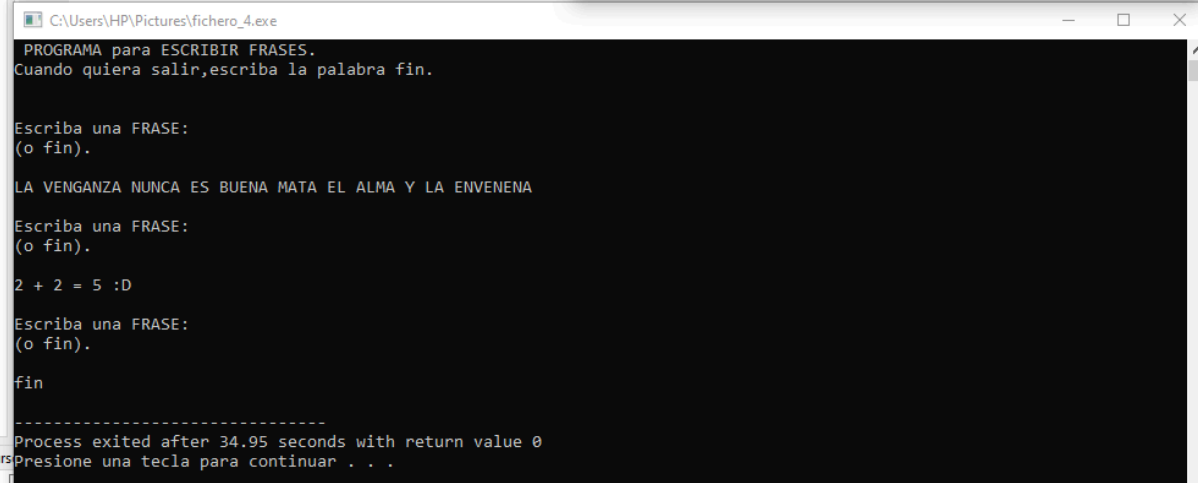
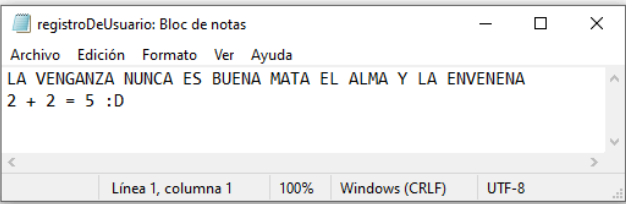
3° Crea un programa en C++ que lea datos desde un archivo de texto llamado "EJEMPLOS.TXT" que fue creado previamente. Este archivo contiene información sobre el desempeño de varios estudiantes, y el programa debe extraer y mostrar esta información en la consola:

```
1  #include <fstream> // Biblioteca para el manejo de ficheros
2  #include <iostream>
3
4  using namespace std; // Biblioteca para la entrada-salida estandar
5  typedef char TCadena [30];
6  int main(){
7      int i;
8      float r;
9      TCadena cad;
10     ifstream fichin("EJEMPLOS.TXT"); // declaracion y apertura del fichero
11     if (!fichin)
12         cout << "\n Incapaz de crear o abrir el fichero ";
13     else{
14         fichin >> i; // Observese la lectura adelantada!!!
15         while (!fichin.eof()){
16             cout << i << " "; // Lectura de valores en el fichero
17             fichin >> r;
18             cout << r << " "; // Lectura de valores en el fichero
19             fichin >> cad;
20             cout << cad << "\n"; // Lectura de valores en el fichero
21             fichin >> i;
22         }
23         fichin.close();
24     } // Fin del main
25 }
```



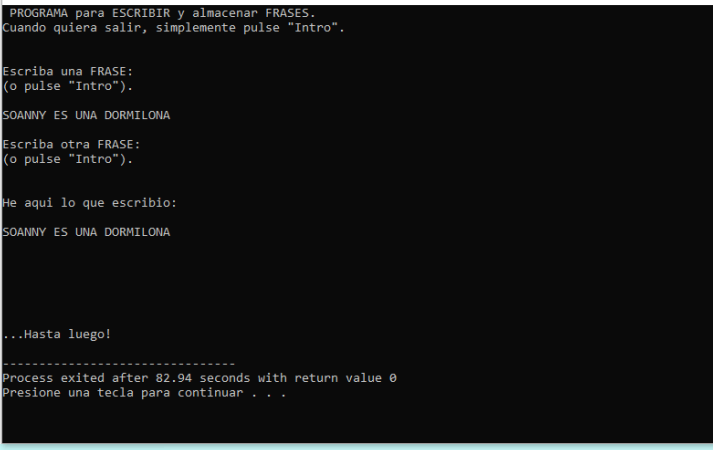
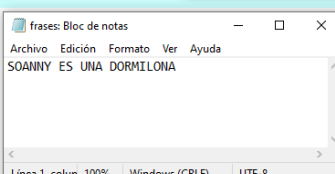
4° Crea un programa en C++ que permita al usuario escribir frases y guardarlas en un archivo de texto llamado "registroDeUsuario.txt". El programa continuará solicitando frases hasta que el usuario ingrese la palabra "fin", momento en el cual se detendrá la entrada y cerrará el archivo.

```
1 #include <stdio.h>
2 #include <string.h>
3 int main()
4 {
5     FILE* ptFichero;
6     char fin[]="fin";
7     char frase [60];
8
9     ptFichero = fopen("registroDeUsuario.txt", "wt");
10    printf(" PROGRAMA para ESCRIBIR FRASES.\nCuando quiera salir,"
11           "escriba la palabra fin.\n\n");
12    do
13    {
14        puts("\nEscriba una FRASE:\n(o fin). \n");
15        gets (frase);
16        if (strcmp(frase, fin) == 0)
17            break;
18        fprintf(ptFichero, "%s\n", frase);
19    }
20    while (strcmp(frase, fin) != 0);
21    fclose(ptFichero);
22    return 0;
23 }
24
```



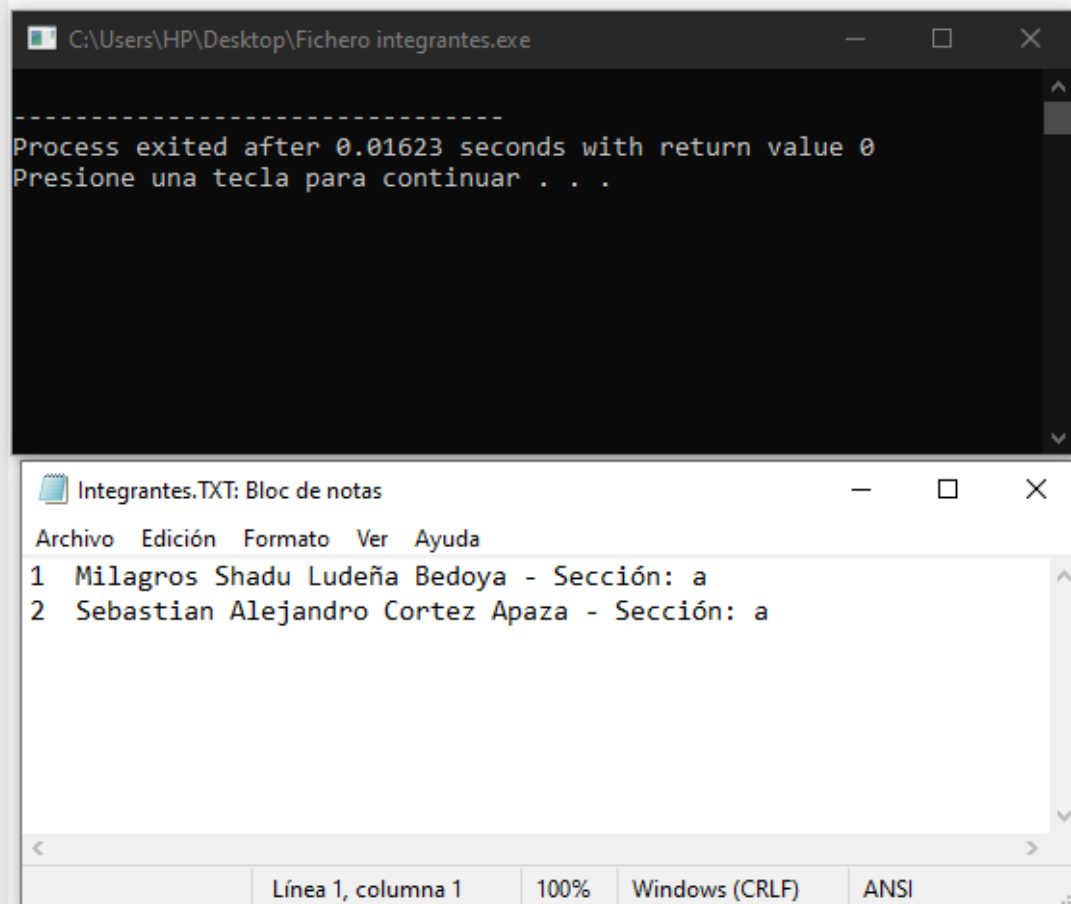
5° Crear un programa en C + + que permita al usuario escribir varias frases y almacenarlas en un archivo de texto llamado "frases.txt". El programa continuará solicitando frases hasta que el usuario presione la tecla "Intro" sin ingresar ningún texto. Después de finalizar la entrada, el programa mostrará todas las frases que el usuario ha escrito.

```
fichero_1.cpp fichero_2.cpp fichero_3.cpp fichero_4.cpp fichero_5.cpp
1 #include <stdio.h>
2 #include <string.h>
3 int main()
4 {
5     FILE* ficheroU;
6     char frase [61];
7     int i=0;
8     ficheroU = fopen("frases.txt", "wt");
9     printf(" PROGRAMA para ESCRIBIR y almacenar FRASES.\n"
10           "Cuando quiera salir, simplemente pulse \"Intro\".\n\n");
11    do
12    {
13        if (i == 0)
14            puts("\nEscriba una FRASE:\n(o pulse \"Intro\"). \n");
15        else
16            puts("\nEscriba otra FRASE:\n(o pulse \"Intro\"). \n");
17        gets(frase);
18        fprintf(ficheroU, "%s\n", frase);
19        i++;
20    }
21    while (strcmp(frase, "") != 0);
22    printf("He aqui lo que escribio:\n\n");
23    fclose(ficheroU);
24
25    ficheroU = fopen("frases.txt", "rt");
26    do
27    {
28        fgets(frase, 60, ficheroU);
29        puts(frase);
30    }
31    while (!feof(ficheroU));
32    getchar();
33    printf("...Hasta luego!");
34    getchar();
35    fclose(ficheroU);
36    return 0;
37
```



6° Crear un programa que guarde en un archivo el nombre y sección de las 3 personas del grupo que hacen esta experiencia práctica.

```
#include <fstream>
#include <iostream>
using namespace std;
int main(){
    setlocale(LC_ALL, "");
    ofstream fichout("Integrantes.TXT", ios:: out);
    if (!fichout)
        cout << "\n Incapaz de crear este o abrir el fichero \n";
    else {
        fichout << 1 << "   Milagros Shadu Ludeña Bedoya - Sección: a" << endl;
        fichout << 2 << "   Sebastian Alejandro Cortez Apaza - Sección: a " << endl;
        fichout.close();
    }
}
```



## Ejercicio - Árboles:

Desarrollar un programa en C++ que implemente un Árbol Binario de Búsqueda (ABB). El programa debe permitir al usuario insertar nodos en el árbol y realizar diferentes recorridos (preorden, en orden y postorden) para mostrar los elementos almacenados.

```
1  #include <iostream>
2  #include <stdlib.h>
3  using namespace std;
4
5  struct nodo{
6      int nro;
7      struct nodo *izq, *der;
8  };
9
10 typedef struct nodo *ABB;
11 /* es un puntero de tipo nodo que hemos llamado ABB, que utilizaremos para mayor facilidad de creacion de variables */
12
13 ABB crearNodo(int x)
14 {
15     ABB nuevoNodo = new(struct nodo);
16     nuevoNodo->nro = x;
17     nuevoNodo->izq = NULL;
18     nuevoNodo->der = NULL;
19     return nuevoNodo;
20 }
21
22 void insertar(ABB &arbol, int x)
23 {
24     if(arbol==NULL)
25     {
26         arbol = crearNodo(x);
27     }
28     else if(x < arbol->nro)
29         insertar(arbol->izq, x);
30     else if (x > arbol->nro)
31         insertar(arbol->der, x);
32 }
33
34 void preOrden(ABB arbol)
35 {
36     if(arbol!=NULL)
37     {
38         cout << arbol->nro << " ";
39         preOrden(arbol->izq);
40         preOrden(arbol->der);
41     }
42 }
43
44 void enOrden(ABB arbol)
45 {
46     if(arbol!=NULL)
47     {
48         enOrden(arbol->izq);
49         cout << arbol->nro << " ";
50         enOrden(arbol->der);
51     }
52 }
53
54 void postOrden(ABB arbol)
55 {
56     if(arbol!=NULL)
57     {
58         postOrden(arbol->izq);
59         postOrden(arbol->der);
60         cout << arbol->nro << " ";
61     }
62 }
63
64 void verArbol(ABB arbol, int n)
65 {
66     if(arbol==NULL)
67         return;
68     verArbol(arbol->der, n+1);
69
70     for(int i=0; i<n; i++)
71         cout<<" ";
72
73     cout<< arbol->nro <<endl;
74
75     verArbol(arbol->izq, n+1);
76 }
77
78 int main()
79 {
80     ABB arbol = NULL; // creado Arbol
81     int n; // numero de nodos del arbol
82     cout << "\n\t\t..[ ARBOL BINARIO DE BUSQUEDA ].. \n\n";
83     cout << " Numero de nodos del arbol: ";
84     cin >> n;
85     cout << endl;
86     for(int i=0; i<n; i++)
87     {
88         cout << " Numero del nodo " << i+1 <<": ";
89         cin >> x;
90         insertar( arbol, x);
91     }
92     cout << "\n Mostrando ABB \n\n";
93     verArbol(arbol, 0);
94     cout << "\n Recorridos del ABB";
95     cout << "\n\n En orden";
96     enOrden(arbol);
97     cout << "\n\n Pre Orden"; preOrden(arbol);
98     cout << "\n\n Post Orden: ";
99     postOrden(arbol);
```

```
95     postOrden(arbol);
96
97     cout << endl << endl;
98
99     system("pause");
100     return 0;
101 }
```

```
C:\Users\HP\Desktop\ejercicio arboles.exe

..[ ARBOL BINARIO DE BUSQUEDA ]..

Numero de nodos del arbol: 2

Numero del nodo 1:2
Numero del nodo 2:3

Mostrando ABB

3
2

Recorridos del ABB

En orden: 2 3

Pre Orden: 2 3

Post Orden: 3 2

Presione una tecla para continuar . . .
```