

Guía práctica de estudio 13: LECTURA Y ESCRITURA DE ARCHIVOS

Codificar en lenguaje C lo siguientes ejercicios:

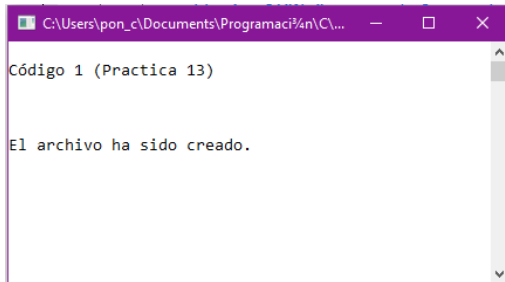
1. Lectura de un archivo de texto carácter por carácter, guardar el contenido en arreglo de caracteres, mostrarlo de forma inversa y guardar de esa forma en otro archivo.

CÓDIGO EN C

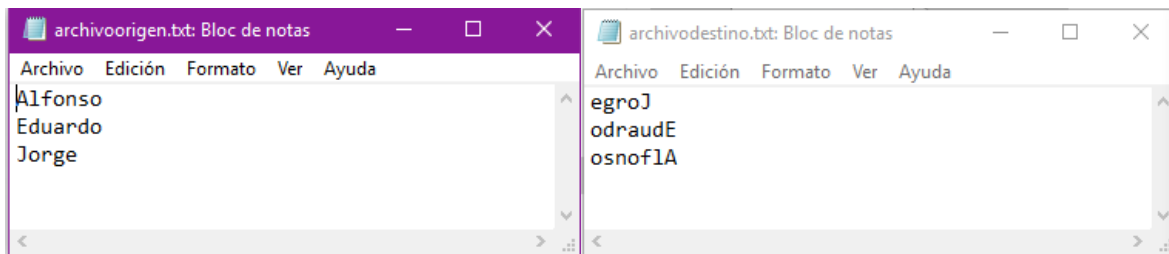
```
1  #include <stdio.h>
2  #include <stdlib.h>
3
4  int main(int argc, char* argv[])
5  {
6      FILE *archivo;
7      FILE *archivo_destino;
8      char contador1, contador2;
9      char c='\0';
10     char arreglo[30]='\0';
11
12     printf("===== \n\n");
13     printf("Código 1 (Practica 13) \n\n", 162);
14
15     if((archivo=fopen("archivoorigen.txt", "r"))==NULL || (archivo_destino=fopen("archivodestino.txt", "w"))==NULL)
16     {
17         printf("No pudo cargar el archivo");
18         getchar();
19         exit(0);
20     }
21
22     else
23     {
24         c = fgetc(archivo);
25         while(!feof(archivo))
26         {
27             arreglo [contador1]=c;
28             //printf("%c", arreglo[contador1]);
29             contador1++;
30             c=fgetc(archivo);
31         }
32         arreglo[contador1]='\0';
33         fclose(archivo);
34
35         printf("\n\n");
36
37         for(contador1=30; contador1>=0 ; contador1-- )
38         {
39             //printf("%c", arreglo[contador1]);
40             fprintf(archivo_destino, "%c", arreglo[contador1]);
41         }
42
43         fclose(archivo_destino);
44         printf("El archivo ha sido creado.");
45
46         getchar();
47         return 0;
48     }
49 }
```

PRUEBA DE ESCRITORIO Y EJECUCIÓN

Programa:



Prueba de escritorio (Archivos de texto, del lado derecho el archivo origen, del lado derecho el archivo destino)



2. Leer con formato dos archivos con una matriz de 3x3 cada uno separando cada columna por espacios y guardar en otro archivo la suma de las matrices con el mismo formato.

CÓDIGO EN C

```
1  #include <stdio.h>
2  #include <stdlib.h>
3  #define MAX 3
4
5  int main(int argc, char **argv)
6  {
7      FILE *archivo=NULL;
8      FILE *archivo2=NULL;
9      FILE *archivo_destino;
10     int contador1, contador2;
11     int matrizUno[MAX][MAX];
12     int matrizDos[MAX][MAX];
13     int sumatrix[MAX][MAX];
14
15     if((archivo=fopen("matrizUno.txt", "r"))==NULL || (archivo_destino=fopen("matrizfinal_destino.txt", "w"))==NULL)
16     {
17         printf("ERROR AL ABRIR ARCHIVO\n");
18         getchar();
19         exit(0);
20     }
21     else
22     {
23         for(contador1=0; contador1<MAX; contador1++)
24             for(contador2=0; contador2<MAX; contador2++)
25             {
26                 fscanf(archivo, "%i", &matrizUno[contador1][contador2]);
27             }
28         fclose(archivo);
29     }
30     //ARCHIVO 2
```

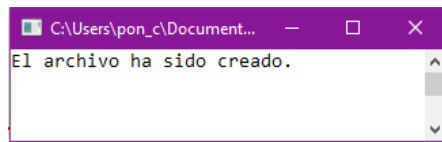
```

31     if((archivo2=fopen("matrizDos.txt","r"))==NULL )
32     {
33         printf("ERROR AL ABRIR ARCHIVO 2\n");
34         getchar();
35         exit(0);
36     }
37     else
38     {
39         for(contador1=0;contador1<MAX;contador1++)
40             for(contador2=0;contador2<MAX;contador2++)
41             {
42                 fscanf(archivo2, "%i", &matrizDos[contador1][contador2]);
43             }
44         fclose(archivo2);
45     }
46     //SUMA E IMPRESION EN EL ARCHIVO FINAL
47     for(contador1=0;contador1<MAX;contador1++)
48     {
49         for(contador2=0;contador2<MAX;contador2++)
50         {
51             sumatrix[contador1][contador2]= matrizUno[contador1][contador2] + matrizDos[contador1][contador2];
52             fprintf(archivo_destino, "%d ", sumatrix[contador1][contador2]); //No poner el &
53         }
54         fprintf(archivo_destino, "\n");
55     }
56     fclose(archivo_destino);
57     printf("El archivo ha sido creado.");
58     getchar();
59     return 0;
60 }

```

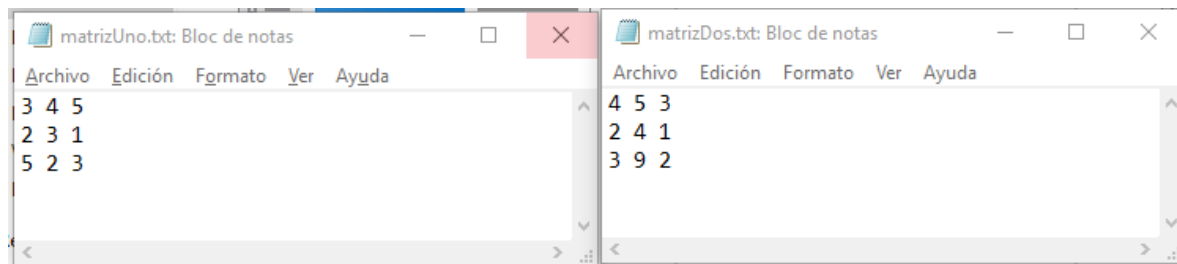
PRUEBA DE ESCRITORIO Y EJECUCIÓN

Programa

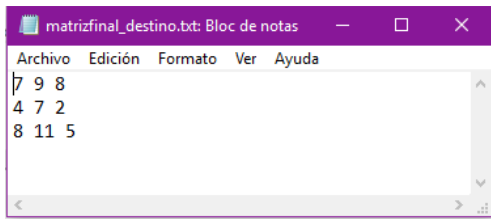


Prueba de escritorio (Archivos de texto, del lado derecho el archivo origen, del lado derecho el archivo destino)

Archivos con las matrices originales



Archivo final con la suma de matrices



3. Leer un conjunto de datos de una columna de un archivo, guardarlo en un arreglo y evaluar cada una en la función $y(x) = a \cdot x \cdot x + b \cdot x + c$ donde los coeficientes a , b y c son dados por el usuario. Mostrar en pantalla los datos recibidos y el resultado de la función para cada uno de los datos, después enviar a un archivo de hoja de cálculo *.xls la columna de valores " x " y en la columna de enfrente el correspondiente valor de " y "

CÓDIGO EN C

```

1  #include <stdio.h>
2  #include <stdlib.h>
3  #define MAX 5
4
5  int main(int argc, char **argv)
6  {
7      FILE *archivo=NULL;
8      FILE *archivo_destino;
9      int contador1, contador2;
10     int matrizUno[MAX];
11     int operacionmatrix[MAX];
12     int coefa, coefb, coefc;
13
14
15     if((archivo=fopen("datosinicio.txt", "r"))==NULL || (archivo_destino=fopen("datosdestino.xls", "w"))==NULL)
16     {
17         printf("ERROR AL ABRIR ARCHIVO\n");
18         getchar();
19         exit(0);
20     }
21     else
22     {
23         for(contador1=0; contador1<MAX; contador1++)
24         {
25             fscanf(archivo, "%i", &matrizUno[contador1]);
26         }
27         fclose(archivo);
28     }
29
30     printf("Este programa sirver para realizar la siguiente ecuaci%c\n", 162);

```

```

31     printf("=====\n\n");
32     printf("NOTA: Los valores de x est%cn en el .txt\n",160);
33     printf("\ty= A*x*x + B*x + C\n");
34     printf("\nIngrese los valores de A, B y C\n");
35     printf("\nIngrese el coeficiente A: ");
36     scanf("%i", &coefa);
37     printf("\nIngrese el coeficiente B: ");
38     scanf("%i", &coefb);
39     printf("\nIngrese el coeficiente C: ");
40     scanf("%i", &coefc);
41
42     //PARTE DE OPERACIONES
43     for(contador1=0;contador1<MAX;contador1++)
44     {
45         operacionmatrix[contador1]= (coefa*matrizUno[contador1]*matrizUno[contador1])+(coefb*matrizUno[contador1])+coefc;
46     }
47
48     //IMPRESION EN EL ARCHIVO FINAL
49     for(contador1=0;contador1<MAX;contador1++)
50     {
51         fprintf(archivo_destino,"%d \t", matrizUno[contador1]); //No poner el &
52         fprintf(archivo_destino,"%d \n", operacionmatrix[contador1]);
53     }
54
55     fclose(archivo_destino);
56     printf("\nEl archivo ha sido creado.");
57     getchar();
58     getchar();
59     return 0;
60 }

```

PRUEBA DE ESCRITORIO Y EJECUCIÓN

Programa

```

C:\Users\pon_c\Documents\Programaci3n\C\Fundamentos de progra...
Este programa sirver para realizar la siguiente ecuaci3n
=====
NOTA: Los valores de x est3n en el .txt
      [y= A*x*x + B*x + C]

Ingrese los valores de A, B y C
Ingrese el coeficiente A: 2
Ingrese el coeficiente B: 4
Ingrese el coeficiente C: 34
El archivo ha sido creado.

```

Prueba de escritorio (Archivos de texto, del lado derecho el archivo origen, del lado derecho el archivo destino)

Archivos con los datos de x

```

datosinicio.txt: Bloc de notas
Archivo  Edici3n  Formato  Ver  Ayuda
5
12
22
34
56

```

Archivo final (.xls) con llos datos de x y los datos ya realizados de y

	A	B	C	D	E
1	5	104			
2	12	370			
3	22	1090			
4	34	2482			
5	56	6530			
6					
7					
8					

Pruebas de escritorio

1) Código 1

Datos ingresados

Alfonso

Datos de Salida

osnofIA

2) Código 2

Matriz 1

3 4 5

Matriz 2

4 5 3

Matriz Suma

7 9 8

3) Código 3

<i>Valor de x</i>	<i>Valor de A</i>	<i>Valor de B</i>	<i>Valor de C</i>
-------------------	-------------------	-------------------	-------------------

12	2	4	34
----	---	---	----

Parte de operaciones

$(2 * 12 * 12) + (4 * 12) + 34$

Valor de y

370

CONCLUSIONES

Murrieta Villegas Alfonso

En la presente práctica mediante el manejo de archivos de texto tanto por la parte de lectura como escritura pudimos realizar una amplia variedad de problemas, desde el simple caso de abrir y cerrar archivos, hasta guardar datos en arreglos para posteriormente utilizarlos y operarlos en matrices, esto posible mediante funciones como `fopen()`, `fclose()`, `fprintf()`, etc. Además, no solo creamos archivos de txt sino que mediante el nombrar la extensión del archivo también pudimos crear tablas de datos en Excel.

Córdova Cervantes Sebastián

En la práctica se vio cómo es posible la lectura y escritura de archivos de texto, a través de código en c. En la pantalla del compilador es posible abrir archivos con formato .txt ya sea únicamente para lectura o para modificarlos.

También se vieron comandos nuevos, tales como el “`fopen()`” o el “`feof()`”