



ASIGNATURA:	ESTUDIANTE:	Quezada Baque Kelvin Peter	
FUNDAMENTOS DE PROGRAMACIÓN NRC 6325	NÚMERO DE LISTA:	25	
	NIVEL/PARALELO:	Primero A	FECHA:03 septiembre 2021
	DOCENTE:	ING. VERÓNICA MARTÍNEZ C. MGS	

## TEMA: EVALUACIÓN TDA Y ARCHIVOS

### Objetivos:

#### General:

Que el estudiante desarrolle la habilidad de crear algoritmos y codificar programas que solucionen problemas específicos aplicando buenas prácticas de programación con la teoría aprendida durante la Unidad3.

#### Específicos:

1. Plantear la solución con la elaboración de un algoritmo que contenga funciones, arreglos y algoritmos de ordenamiento y búsqueda.
2. Desarrollar el algoritmo y documentar la información a detalle en el presente informe.
3. Adjuntar la carpeta del proyecto **#U3Eva2ApellidoNombre**

### Ejercicio:

1. Crear la estructura del programa, según lo expuesto por la docente.
2. Documentar el programar en el **formato del presente informe:**
  1. Código del programa en C++
  2. Captura de código en C++
  3. Captura de ejecución en C++
3. Entregar una carpeta comprimida con el nombre **#U3Eva2ApellidoNombre** que contenga:
  1. **#U3Eva2ApellidoNombre** este DOCUMENTO en formato pdf.
  2. **#U3Eva2ApellidoNombre** Carpeta completa del proyecto en C++

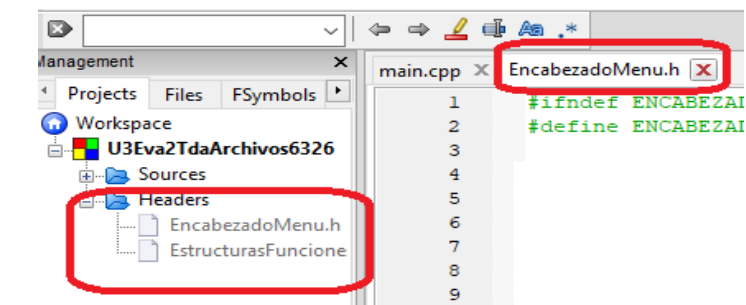
**Ejercicio:**

Desarrollar un algoritmo y programa usando todo lo visto en la Unidad 3. El mismo contendrá: **TODAS LAS VARIABLES Y PROCEDIMIENTOS CON SUS INICIALES DE APELLIDONOMBRE STRING, INT, FLOAT, CHAR, DOUBLE**

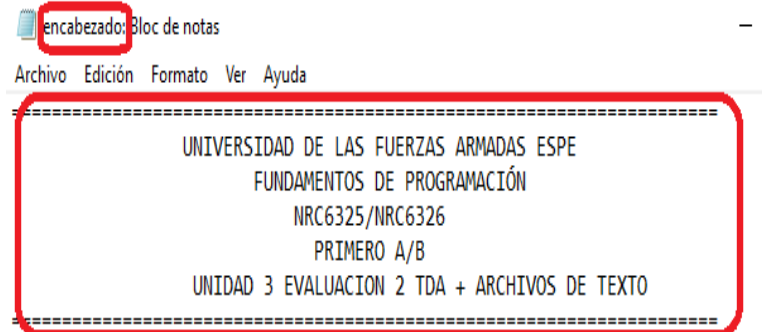
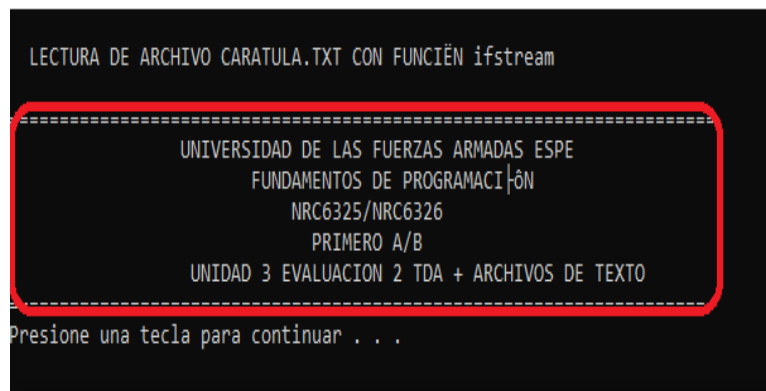
**Algoritmo:** Debe realizar el algoritmo paso a paso a detalle cumpliendo con lo siguiente:

- 1) Crear una Librería ENCABEZADOMENU: Que cargará la información desde un Encabezado.txt del encabezado y menú.txt del menú.

Crear una librería para ENCABEZADO Y MENU



"D:\Ejercicios CodeBlocks\6325\Unidad3\U3Eva2TdaArchivos6326\bin\Debug\U3Eva2TdaArchivos6326.exe"





- 2) Crear una Librería **ESTRUCTURASFUNCIONES**: Que contiene:
- 1 estructura simple con 3 variables de diferentes tipos de datos.
  - 1 estructura compuesta con 2 variables de diferentes tipos de datos y 1 variable objeto de tipo TDA SIMPLE.

El menú tiene los datos:

1. Crear e Imprimir Archivo con Struct Simple con **DATOS INGRESADOS**
2. Crear e Imprimir Archivo con Struct Simple/Compuesta con **DATOS INGRESADOS**
3. Salir

**Case 1:**

**StructSimple(); REALIZA Y CREA EL TDA DATOS: INT,  
STRING,FLOAT  
MostrarStructSimple();**

**Case 2:**

**StructCompuesta();  
MostrarStructCompuesta(); DATOS: CHAR, FLOAT**

#### **ESTRUCTURA DEL PROGRAMA:**

1. Librería 1 **ENCABEZADOMENU**.
  - a. Cargar desde un txt el archivo encabezado.
  - b. Cargar desde un txt el archivo menú.
2. Librería 2 **ESTRUCTURASFUNCIONES**
  - a. Funciones para crear TDA SIMPLE. 3 VARIABLES.
  - b. Funciones crear e imprimir TDA SIMPLE Y COMPUESTA. 2 VARIABLES.
3. Funciones del programa principal MAIN (), LLAMAR A ENCABEZADO Y MENU.



"D:\Ejercicios CodeBlocks\6325\Unidad3\U3Eva2TdaArchivos6326\bin\Debug\U3Eva2TdaArchivos6326.exe"

Ingresar opción: 1

PASO: 1

=====

TDA SIMPLE CON DATOS ALMACENADOS EN ARCHIVO StructSimple

=====

----- Docente 1 -----

Docente : Anita Ramirez  
id : 123  
Sueldo :449.85  
Precio Hora : 52  
Profesión : Finanzas

**INGRESADOS Y  
GUARDADOS**

----- Docente 2 -----

Docente : Fabricio Ramirez  
id : 456  
Sueldo :228.15  
Precio Hora : 50  
Profesión : Administración

Presione una tecla para continuar . . .

PASO: 2

\*\*\*\*\* COMPROBANDO LECTURA CON ifstream DE LA STRUCT SIMPLE \*\*\*\*\*  
ASIGNADA DESDE ARCHIVO CREADO

=====

DESDE ARCHIVO TDA SIMPLE CON DATOS ALMACENADOS EN ARCHIVO StructSimple

=====

----- Docente 1 -----

Docente : Anita Ramirez  
id : 123  
Sueldo :449.85  
Precio Hora : 52  
Profesión : Finanzas

**CARGADOS  
DESDE EL  
ARCHIVO TXT**

----- Docente 2 -----

Docente : Fabricio Ramirez  
id : 456  
Sueldo :228.15  
Precio Hora : 50  
Profesión : Administración

StructSimple: Bloc de notas

Archivo Edición Formato Ver Ayuda

=====

DESDE ARCHIVO TDA SIMPLE CON DATOS ALMACENADOS EN ARCHIVO StructSimple

=====

----- Docente 1 -----

Docente : Anita Ramirez  
id : 123  
Sueldo :449.85  
Precio Hora : 52  
Profesión : Finanzas

----- Docente 2 -----

Docente : Fabricio Ramirez  
id : 456  
Sueldo :228.15  
Precio Hora : 50  
Profesión : Administración



Ingresar opción: 2

```
PASO 1: CARGANDO EL STRUCT COMPUESTO
        TDA COMPUESTA CON INGRESO DE DATOS POR CONSOLA
```

Ingresar número de docentes: 2

```
----- Docente 1 -----
1. Nombre      : Luis Armando
2. Apellido    : Ortiz
3. Materia     : Estructuras
4. Número Horas : 8
----- Docente 2 -----
1. Nombre      : Margoth
2. Apellido    : Guaraca
3. Materia     : P00
4. Número Horas : 6
Presione una tecla para continuar . . .
```

PASO: 2

```
***** COMPROBANDO LECTURA CON ifstream DE LA STRUCT COMPUESTA *****
        CON DATOS INGRESADOS DESDE ARCHIVO CREADO
```

```
PASO: 2 EL ARCHIVO TIENE:
        TDA COMPUESTA CON INGRESO DE DATOS POR CONSOLA
```

```
----- Docente 1 -----
1. Nombre      : Luis Armando
2. Apellido    : Ortiz
3. Materia     : Estructuras
4. Número Horas : 8
----- Docente 2 -----
1. Nombre      : Margoth
2. Apellido    : Guaraca
3. Materia     : P00
4. Número Horas : 6
```

StructCompuestaIngreso: Bloc de notas

Archivo Edición Formato Ver Ayuda

```
PASO: 2 EL ARCHIVO TIENE:
        TDA COMPUESTA CON INGRESO DE DATOS POR
```

```
----- Docente 1 -----
1. Nombre      : Luis Armando
2. Apellido    : Ortiz
3. Materia     : Estructuras
4. Número Horas : 8
```

```
----- Docente 2 -----
1. Nombre      : Margoth
2. Apellido    : Guaraca
3. Materia     : P00
4. Número Horas : 6
```

Línea 1, columna 1 100% UNIX (LF) ANSI



## Desarrollo:

1. **Captura de código en C++**
  - a. Librería 1 Estructura

```
#include <iostream>
#include <stdlib.h>
#include <fstream>
#include <wchar.h>
#include <locale.h>
#include <wchar.h>
#include "encabezadoMenuKPQB.h"
#include "estructuraFuncionKPQB.h"
```

```
using namespace std;
void menuPrincipalKQ();
```

```
int main(){
    menuPrincipalKQ();

    return 0;
}
```

- b. Librería 2 Menú Principal

```
#ifndef ENCABEZADOMENUKPQB_H_INCLUDED
#define ENCABEZADOMENUKPQB_H_INCLUDED
using namespace std;
void menuPrincipalKQ();
```



```
void datosEmpleados();
void menuPrincipalKQ();
void artefactosEmpresa();
void menuPrincipalKQ(){
    int opcionMenu;
    do{
        cout<<"===== "<<endl;
        cout<<"          UNIDAD 3 TALLER 1 TDA + ARCHIVOS  " <<endl;
        cout<<"===== "<<endl;
        cout<<"===== "<<endl;
        cout<<"          MENU PRINCIPAL TDA + ARCHIVOS  " <<endl;
        cout<<"===== "<<endl;
        cout<<"1. Crear e Imprimir Archivo con Struct Simple con DATOS INGRESADOS " <<endl;
        cout<<"2. Crear e Imprimir Archivo con Struct Simple/Compuesta con DATOS" <<endl;
        cout<<"3. Salir " <<endl;
        cout<<"===== "<<endl;
        cout<<" Escoja una de las 4 opciones " <<endl;
        cout<<"===== "<<endl;
        cin >> opcionMenu;
        switch(opcionMenu){
            case 1:

                datosEmpleados();

                break;
            case 2:
                artefactosEmpresa();

                break;
            case 3:
                cout << "GRACIAS POR VENIR VUELVA PRONTO" << endl;
                exit(0);
                break;
            default:
                cout <<"===== " <<endl;
                cout <<"OPCION INCORRECTA, INGRESE NUEVAMENTE" <<endl;
        }
    }
    while (!((opcionMenu<0 && opcionMenu>=3)));
}

#endif // ENCABEZADOMENUKQB_H_INCLUDED
```



c. Funciones del programa principal

```
#ifndef ESTRUCTURAFUNCIONKPQB_H_INCLUDED
#define ESTRUCTURAFUNCIONKPQB_H_INCLUDED
using namespace std;

void lecturaArchivo();
void datoArchivo();

struct informacionEmpleados{ //estructura simple
    string nombre;
    string apellido;
    char area[40];
    int numEmpl;
}empleados;

struct artefacto{ //simple
    char direccion[30];
    char color[20];
    char marca[20];
};

struct empleado{ //anidada
    char nombreCliente[20];
    char area[20];
    char nombre[20];
    struct artefacto infEmpleado;
    double salario;
}empleados2[2];

void artefactosEmpresa(){
    int numero,i;
    cout<<"===== "<<endl;
    cout<<"    ingrese los artefactos"<<endl;
    cout<<"===== "<<endl;
    cout<<"Ingrese cuantas atefactos va a registrar "<<endl;
    cin>>numero;
    while (numero<=0) {
        cout << "No se pueden ingresar numeros negativos" << endl;
        cout << "Por favor intente de nuevo" << endl;
        cin >> numero;
    }
}
```





```
}

for (int i=0;i<numero;i++){
cout<<"===== "<<endl;
cout<<"Digite el nombre y apellido: ";
fflush(stdin);
cin.getline(empleados2[i].nombreCliente,20,'\n');
cout<<"Digite el Area: ";
cin.getline(empleados2[i].area,20,'\n');
cout<<"Digite la direccion: ";
cin.getline(empleados2[i].infEmpleado.direccion,30,'\n');
cout<<"Digite el nombre del artefacto: ";
cin.getline(empleados2[i].nombre,20,'\n');
cout<<"Marca: ";
cin.getline(empleados2[i].infEmpleado.marca,20,'\n');
cout<<"Color: ";
cin.getline(empleados2[i].infEmpleado.color,20,'\n');
cout<<"Precio: ";
cin>>empleados2[i].salario;
while (empleados2[i].salario<=0) {
cout << "No se pueden ingresar numeros negativos" << endl;
cout << "Por favor intente de nuevo" << endl;
cin >>empleados2[i].salario;
cout<<"===== "<<endl;
}

}

cout<<"\n";

}

void datoArchivo2(){
int i;
ofstream datos2; //ficheiros de salida
datos2<<"----- "<<endl;
datos2.open("RegistroCompra.txt", ios::out | ios::binary);
datos2<<"SU REGISTRO SON " <<endl;
datos2<<" NOMBRE Y APELLIDO: "<<empleados2[i].nombreCliente<<endl;
datos2<<" AREA: "<<empleados2[i].area<<endl;
datos2<<" DIRECCION: "<<empleados2[i].infEmpleado.direccion<<endl;
datos2<<" ARTEFACTO: "<<empleados2[i].nombre<<endl;
datos2<<" MARCA: "<<empleados2[i].infEmpleado.marca<<endl;
datos2<<" COLOR: "<<empleados2[i].infEmpleado.color<<endl;
datos2<<" PRECIO: "<<empleados2[i].salario<<endl;
datos2<<"----- "<<endl;
lecturaArchivo();
}
```



```
void datosEmpleados(){
    cout<<"\n\t--- Registro de empleados --- "<<endl;
    cout<<"===== "<<endl;
    cout << "Ingrese el nombre: ";
    cin>>empleados.nombre;
    cout<<"Ingrese el apellido: ";
    cin>>empleados.apellido;
    cout<<"Ingrese su area  : ";
    cin.ignore();
    cin.getline(empleados.area,20,'\n');
    cout<<"Ingrese su codigo : ";
    cin>>empleados.numEmpl;
    datoArchivo2();
}

void datoArchivo(){
    ofstream datos; //ficheiros de salida
    datos<<"----- "<<endl;
    datos.open("Registro.txt", ios::out | ios::binary);
    datos<<"SU REGISTRO SON "<<endl;
    datos<<" NOMBRE: "<<empleados.nombre<<endl;
    datos<<" APELLIDO: "<<empleados.apellido<<endl;
    datos<<" AREA: "<<empleados.area<<endl;
    datos<<" CODIGO: "<<empleados.numEmpl<<endl;
    datos<<"----- "<<endl;
    lecturaArchivo();
}

void lecturaArchivo(){
    cout<<"|| LECTURA DE ARCHIVO ||"<<endl<<endl;
    ifstream lectura;
    string textoLectura;
    lectura.open("Registro.txt", ios::in | ios::binary);
    while (!lectura.eof()){//determina el fin de un archivo
        getline(lectura,textoLectura);
        cout<<textoLectura<<endl;
    }
}
```



**ESPE**  
UNIVERSIDAD DE LAS FUERZAS ARMADAS  
INNOVACIÓN PARA LA EXCELENCIA

**Sede  
Santo Domingo**

```
}  
lectura.close(); //abrir y cerrar ficheros  
cout<<endl<<endl;  
system("pause");//genera pausa en el programa  
}  
  
#endif // ESTRUCTURAFUNCIONKPQB_H_INCLUDED
```



## 2. Captura de ejecución en C++

a. Carátula

b. Menú

```
=====
                        MENU PRINCIPAL TDA + ARCHIVOS
=====
1. Crear e Imprimir Archivo con Struct Simple con DATOS INGRESADOS
2. Crear e Imprimir Archivo con Struct Simple/Compuesta con DATOS
3. Salir
=====
Escoja una de las 4 opciones
=====
1
```

c. Op1. TDA Simple

```
=====
                        UNIDAD 3 TALLER 1 TDA + ARCHIVOS
=====
                        MENU PRINCIPAL TDA + ARCHIVOS
=====
1. Crear e Imprimir Archivo con Struct Simple con DATOS INGRESADOS
2. Crear e Imprimir Archivo con Struct Simple/Compuesta con DATOS
3. Salir
=====
Escoja una de las 4 opciones
=====
1

--- Registro de empleados ---
=====
Ingrese el nombre: kelvin
Ingrese el apellido: quezada
Ingrese su area      : cajero
Ingrese su codigo    : 147
|| LECTURA DE ARCHIVO ||

kelvin
quezada
cajero
478
=====
```



d. Op2. TDA Compuesto

```
=====
                                UNIDAD 3 TALLER 1 TDA + ARCHIVOS
=====
                                MENU PRINCIPAL TDA + ARCHIVOS
=====
1. Crear e Imprimir Archivo con Struct Simple con DATOS INGRESADOS
2. Crear e Imprimir Archivo con Struct Simple/Compuesta con DATOS
3. Salir
=====
Escoja una de las 4 opciones
=====
2
=====
                                ingrese los artefactos
=====
Ingrese cuantas artefactos va a registrar
2
=====
Digite el nombre y apellido: kkk
Digite el Area: eeee
Digite la direccion: www
Digite el nombre del artefacto: compu
Marca: sonyy
Color: blanco
Precio: 150
=====
Digite el nombre y apellido: maria
Digite el Area: vendedora
Digite la direccion: chone
Digite el nombre del artefacto: celular
Marca: hp
Color: negro
```



Firma del Estudiante:

Apellido Nombre: Quezada Baque Kelvin Peter

Cédula: 2350196925