

Universidad Nacional de La Matanza

Programación 1110 - Comisión 05-1965 (Lunes - Jueves - Noche)

Primer Examen Parcial - 23/05/2022

Sección Lenguaje C

1. Desarrolle una función que normalice una cadena de caracteres. Determine el criterio de la normalización a partir de la salida volcada al archivo de salida.
2. Se tienen dos archivos con el stock de un almacén. Se los desea fusionar en un nuevo único archivo binario. Desarrolle una función que resuelva la operación de fusión. Los archivos de entrada se encuentran ordenados por el código de producto. Los archivos deben recorrerse solo UNA vez.
3. Diseñe e implemente una función que cargue el archivo resultante de la fusión generada en el punto anterior en una lista simplemente enlazada.
4. Diseñe e implemente una función que elimine los duplicados de la lista del punto anterior. La eliminación debe ser con acumulación. Sobrescriba el archivo del punto anterior con los registros contenidos en la lista.
5. Se dispone de un archivo con las novedades relacionadas a los productos (ingresos y egresos) de productos. Se solicita actualizar el maestro de stock a partir del archivo de novedades mencionado.

Sección Lenguaje C++

Cada ejercicio debe ser un nuevo proyecto organizado en archivos separados (main.cpp, xxx.h, xxx.cpp).

1. Desarrollar una clase que simule el comportamiento de una cadena de texto, sin utilizar el tipo `string` de C++. La misma se debe poder utilizar con el siguiente main:

```
int main()
{
    Cadena c1("una cadena");
    Cadena c2(c1);
    Cadena c3;
    Cadena c4;

    c2 = "otro ejemplo de cadena";
    c4 = "otra cadena de varios caracteres";

    cout << "Longitud de c1 = " << c1.longitud() << endl;
    cout << "Longitud de c2 = " << c2.longitud() << endl;
    cout << "Longitud de c3 = " << c3.longitud() << endl;
    cout << "Longitud de c4 = " << c4.longitud() << endl;
    return 0;
}
```

2. Desarrollar una clase que simule el comportamiento de una cadena de texto, sin utilizar el tipo `string` de C++. La misma se debe poder utilizar con el siguiente main:

```
int main()
{
    Cadena c1("una cadena"),
```

```
        c2(c1),
        c3(c1 + c2),
        c4, c5, c6;

    c4 = c1 + " de varios caracteres";
    cin >> c5;
    c6 = "Esta es " + c1;

    cout << "c1 = " << c1 << endl
        << "c2 = " << c2 << endl
        << "c3 = " << c3 << endl
        << "c4 = " << c4 << endl
        << "c5 = " << c5 << endl
        << "c6 = " << c6 << endl;

    return 0;
}
```

Condiciones de aprobación del examen

- El programa debe compilar correctamente en todos los casos.
- Debe implementar las funciones finalizadas en “_MIO”. No debe realizar funciones monolíticas.
- La compilación debe estar exenta de **warnings** arrojados por el compilador.
- Las salidas con los resultados del archivo `pantalla.txt` deben ser **iguales** a las que se encuentran en el archivo `pantallaBIEN.txt`.
- La interpretación de los requisitos y la estructura de la aplicación es parte de la evaluación.
- Requisitos para nota **4** (cuatro): puntos anteriores, dos ejercicios a su elección de la Sección Lenguaje C y un ejercicio a su elección de la Sección Lenguaje C++.
- Requisitos para nota **7** (siete): puntos anteriores. Un ejercicio adicional de la Sección Lenguaje C y declaración de las clases de un ejercicio adicional de la Sección Lenguaje C++.

Observaciones generales

- Para el proyecto de la Sección Lenguaje C recuerde tener deshabilitado el flag `-std=c99`.
- Para la Sección Lenguaje C sólo debe entregar los archivos **funciones.c** y **que_hice.h**.
- Para la Sección Lenguaje C++ sólo debe entregar los archivos **.cpp** y **.h** de la clase desarrollada.
- Generar un archivo Zip con los archivos a entregar indicados en los apartados anteriores. El archivo debe tener el nombre:

DDDDDDDD_20220523_parcial11.zip, donde DDDDDDDD es el número de DNI del alumno. Subir el archivo como entrega en la plataforma MIeL.