

# ESCUELA DE INGENIERÍA INFORMÁTICA DE OVIEDO

---

## Fundamentos de Computadores Y Redes

---

Curso 2024-2025

### Trabajo Grupal - Fase I

Díaz Mendaña, Diego - UO301887

García Pernas, Pablo - UO??????

Gota ?, Jorge - UO??????

Suárez Fernández, Fernando - UO300028

**Grupo de prácticas: ?**

**Titulación:** PCEO Informática y Matemáticas

21 de marzo de 2025

# Índice

<b>1. Introducción</b>	<b>2</b>
<b>2. ID</b>	<b>2</b>
<b>3. Desarrollo del programa</b>	<b>2</b>
3.1. Descripción general . . . . .	2
3.2. Funciones implementadas . . . . .	2
3.2.1. ControlWithReversedStrings() . . . . .	2
3.2.2. MaskControl() . . . . .	3
3.2.3. ControlInAsm() . . . . .	3
3.2.4. CheckArray() . . . . .	3
<b>4. Cuestiones</b>	<b>3</b>
4.1. Cuestión 1 . . . . .	3
4.2. Cuestión 2 . . . . .	3
4.3. Cuestión 3 . . . . .	3
4.4. Cuestión 4 . . . . .	3
<b>5. Distribución del trabajo</b>	<b>3</b>

## 1. Introducción

La presente memoria da cuenta del desarrollo de la Fase I del trabajo práctico correspondiente a la asignatura Fundamentos de Computadores y Redes, con el propósito de profundizar en los conceptos generales tratados en los laboratorios de la citada asignatura. A través de la implementación de un sistema de control de licencias parametrizado, se integran conocimientos relativos al lenguaje C/C++ y al ensamblador sobre arquitectura Intel x86-64, abordando aspectos fundamentales como el manejo de bits, máscaras, desplazamientos y estructuras de control.

## 2. ID

Para el desarrollo del presente proyecto se ha seleccionado como identificador de referencia el número *UO300028*, al ser el más bajo entre los disponibles dentro del grupo, conforme a lo estipulado en la guía oficial de la práctica. No obstante, cabe señalar que, en determinados fragmentos del código como se detallará en secciones posteriores de esta memoria, se ha optado por la utilización de identificadores alternativos. Esta decisión ha respondido a la necesidad de preservar la coherencia lógica y funcional del programa, la cual se veía comprometida al emplear un único identificador de forma estricta en todos los contextos del desarrollo.

## 3. Desarrollo del programa

### 3.1. Descripción general

El programa desarrollado consiste en un sistema simplificado de control de licencias, parametrizado mediante un identificador numérico asociado al número de matrícula universitaria. Su implementación combina el lenguaje C++, utilizado para la lógica principal y la gestión de entrada/salida, con código en ensamblador x86-64 para una función específica de validación. Todo el desarrollo se ha realizado en el entorno *Visual Studio*, respetando la configuración proporcionada por la asignatura (*Teamwork*).

### 3.2. Funciones implementadas

#### 3.2.1. `ControlWithReversedStrings()`

La función implementada tiene por objeto la verificación del formato de dos cadenas introducidas por el usuario a través de la consola, conforme a los criterios establecidos en el enunciado de la práctica. Para la inversión de la segunda cadena, se ha recurrido a la función nativa `strrev()`, propia del lenguaje C++, lo que permite una manipulación directa y eficiente de las estructuras de texto. A continuación, se muestran sendos ejemplos de entrada válida e inválida:

- **Entrada válida:** aaaaaaaaaaaaaa y 1df4g7n9
- **Entrada inválida:** a y a

### 3.2.2. `MaskControl()`

La función comprueba el formato de dos enteros recibidos por consola siguiendo las reglas impuestas en el guión de la práctica. (Añadir más información)? A continuación, se muestran sendos ejemplos de entrada válida e inválida:

- **Entrada válida:**
- **Entrada inválida:**

### 3.2.3. `ControlInAsm()`

Blablabla

### 3.2.4. `CheckArray()`

Blablabla

## 4. Cuestiones

A continuación se presentan las respuestas a las cuestiones planteadas en el guión de la práctica.

### 4.1. Cuestión 1

### 4.2. Cuestión 2

### 4.3. Cuestión 3

### 4.4. Cuestión 4

## 5. Distribución del trabajo

La implementación del proyecto se ha organizado siguiendo una estrategia de trabajo colaborativo, basada en la asignación individual de cada una de las funciones principales descritas en el enunciado. Cada integrante del grupo asumió la responsabilidad del desarrollo, depuración y validación de una función específica, lo que favoreció una distribución equilibrada de las tareas y una mayor especialización técnica en cada componente del sistema.

Como soporte para el trabajo en equipo, se ha utilizado el sistema de control de versiones distribuido *Git* en combinación con la plataforma *Github* como repositorio remoto

centralizado. El grupo ha adoptado un flujo de trabajo basado en la estrategia de *feature branching*, en la cual cada funcionalidad fue desarrollada en una rama independiente derivada de la rama principal **main**. Una vez finalizada la implementación de cada funcionalidad, se procedía a la apertura de una **pull request** (PR), a través de la cual se solicitaba la revisión del código por parte de los demás integrantes.

Este enfoque ha permitido establecer un proceso sistemático de *code review*, mediante el cual cada **commit** propuesto era examinado antes de su integración definitiva en la rama principal del repositorio. Asimismo, se han empleado herramientas integradas de GitHub como la resolución de *merge conflicts*, en caso de ser necesarios. Las revisiones cruzadas entre miembros han servido tanto para detectar errores lógicos como para asegurar la consistencia del estilo y la correcta integración de las distintas partes del código.

Gracias a este flujo de trabajo, se ha logrado mantener una trazabilidad completa de los cambios, facilitar la colaboración asíncrona y reforzar la calidad del desarrollo mediante la incorporación de prácticas propias del desarrollo profesional de software.

El repositorio que ha servido como entorno de trabajo colaborativo se encuentra alojado en la plataforma *GitHub*, bajo el siguiente enlace: <https://github.com/PabloGarPe/fcrtrabajo>. Cabe señalar que, en el momento de redacción de esta memoria, el repositorio permanece configurado como privado, por lo que su acceso podría no estar disponible de forma pública. No obstante, se encuentra íntegramente documentado y estructurado conforme a las convenciones adoptadas durante el desarrollo, con un historial completo de *commits*, ramas y solicitudes de fusión que evidencian el flujo de trabajo seguido por el grupo.