



# **Fundamentos de Programación.**

## **Guión de Prácticas.**

Curso 2012/2013

---

Autor: Juan Carlos Cubero

Para cualquier sugerencia o comentario sobre este guión de prácticas, por favor, enviad un e-mail a [JC.Cubero@decsai.ugr.es](mailto:JC.Cubero@decsai.ugr.es)

---

Profesores de prácticas: M.Gómez y S. Acid

Para cualquier cuestión metodológica ponerse en contacto con los profesores o bien enviando un e-mail a [mgomez@decsai.ugr.es](mailto:mgomez@decsai.ugr.es) y [acid@decsai.ugr.es](mailto:acid@decsai.ugr.es).

---

*"Lo que tenemos que aprender a hacer, lo aprendemos haciéndolo".*

*Aristóteles*



*"In theory, there is no difference between theory and practice. But, in practice, there is".*

*Jan L. A. van de Snepscheut*



*"The gap between theory and practice is not as wide in theory as it is in practice".*



*"Theory is when you know something, but it doesn't work. Practice is when something works, but you don't know why. Programmers combine theory and practice: Nothing works and they don't know why".*



## **Sobre el guión de prácticas**

Este guión de prácticas contiene las actividades a realizar por el alumno, tanto de forma presencial (en las aulas de la Escuela de Informática) como no presencial a realizar en casa. Todas las actividades son obligatorias y están incluidas en la evaluación de la participación de clase, excepto las marcadas como *Actividades de Ampliación*.

El guión está dividido en sesiones, que corresponden a cada semana lectiva. Como ya se indicara en clase, para la asignatura de FP a cada semana le corresponden 6 horas de trabajo personal del alumno, aparte de las 4 horas de clases presenciales. Por ello, las actividades no presenciales de una sesión deben desarrollarse **antes**, *durante toda una semana*, y culminan con una entrega de los ejercicios en la propia sesión de prácticas. Por lo que el alumno deberá descargarse al terminar una sesión, las actividades encargadas para la sesión siguiente.

Entre las actividades del guión, una actividad no presencial recurrente es, la resolución de problemas propuestos en las *Relaciones de Problemas*. Éstas constan de dos tipos de problemas:

- **Básicos:**

Deben resolverse en la casa y el alumno deberá defenderlos individualmente. La defensa será individual y forma parte de la nota final de la asignatura (10 % tal y como se explica en el apartado de evaluación de la asignatura).

Las soluciones a los problemas deberán ser subidas a la plataforma de decsai a través del GAP. Para ello, el alumno debe entrar en el acceso identificado de decsai, entrar por el GAP, y seleccionar la entrega de prácticas correspondiente a la semana en curso. El alumno subirá cada uno de los ficheros con extensión `cpp` correspondientes a las soluciones de los ejercicios propuestos (tanto obligatorios, como opcionales) fijados en esa sesión.

**Nota:** Para una correcta gestión de los ficheros en el GAP, los nombres de los ficheros deberán ser de la forma: `solucion1.cpp`, `solucion2.cpp`, `solucion3.cpp`, etc. correspondientes al número de ejercicio con el que figuran en el guión de prácticas.

El profesor entregará al final de cada semana las soluciones a los ejercicios básicos. Es muy importante que el alumno revise estas soluciones y las compare con las que él había diseñado.

Del conjunto de problemas básicos, en el guión de prácticas se distinguirá entre:

1. **Obligatorios:**

Si se realizan correctamente estos ejercicios, el alumno podrá sacar hasta un 9 (sobre 10) de nota.

2. *Opcionales:*

Si se realizan correctamente estos ejercicios, el alumno podrá sacar hasta un 10 (sobre 10) de nota.

- *Ampliación:* problemas cuya solución no se verá, pero que sirven para afianzar conocimientos. El alumno debería intentar resolver por su cuenta un alto porcentaje de éstos.

Las actividades marcadas como *Seminario* han de hacerse obligatoriamente y siempre serán expuestas por algún alumno elegido aleatoriamente.

Para la realización de estas prácticas, se utilizará el entorno de programación Code::Blocks. Se recomienda la instalación del programa lo antes posible (en la primera semana de prácticas). En cualquier caso, el alumno puede instalar en su casa cualquier otro compilador, como por ejemplo Visual Studio → [www.microsoft.com/visualstudio/](http://www.microsoft.com/visualstudio/)

---

*Muy importante:*

- La resolución de los problemas y actividades puede hacerse en grupo, pero la defensa durante las sesiones presenciales es individual.
- Llevar la asignatura al día para poder realizar los ejercicios propuestos para cada sesión.

---

## Sesión 8

---

### ► Actividades a realizar en casa

#### **Funciones y Clases**

##### **Actividad: Resolución de problemas.**

Resolved los ejercicios siguientes de la relación de problemas III de Funciones y Clases. Más concretamente es sobre clases.

- *Obligatorios:*

7 `solucion7.cpp`

8 `solucion8.cpp`

10 `solucion10.cpp`

- *Opcionales:*

- \* Ejercicio fuera de la relación. `solucion26.cpp`

Modifica la clase `Polinomio` de grado 2, definida en el fichero `poli02.cpp` que se encuentra en `decsai ->Material ->Teoría`, para que incluya un método adicional llamado `grado`. Este método debe devolver el grado de un polinomio. Será necesario además, incluir una nueva opción en el menú que habilite la llamada a dicho método desde la función `main()`.

9 `solucion9.cpp`