

## LABORATORIO DE PROGRAMACIÓN II

*Juan Hernández Núñez  
Fernando Sánchez Figueroa  
Luis Álvarez Llorente  
José Luis Herrero Agustín  
Adolfo Lozano Tello  
Jorge Quirós Rosado*

*e-mail: {juanher, fernando, luisalva, jherrero, alozano, jquiros}@unex.es*  
Departamento de Informática. Escuela Politécnica. Universidad de Extremadura.

### **RESUMEN**

*La asignatura de Laboratorio de Programación II es complementaria de Laboratorio de Programación I; se pretende que el alumno alcance un grado mínimo en la programación de lenguajes orientados a objetos. Exponemos aquí una visión general de la asignatura con el temario y la problemática que encontramos en la didáctica de la misma.*

El objetivo de la asignatura es el aprendizaje de las técnicas de programación orientadas a objetos usando los lenguajes PASCAL orientado a objetos y C++. El aprendizaje se lleva a cabo en salas de prácticas de forma tutelada.

Se aprovechará el conocimiento de programación en PASCAL que los alumnos han adquirido en la asignatura de Laboratorio de Programación I. Se pedirá para aprobar la asignatura que los alumnos desarrollen individualmente una práctica en PASCAL, utilizando programación orientada a objetos, con las especificaciones expuestas en un enunciado. Además se requiere que el alumno domine la programación orientada a objetos en C++.

El número de alumnos matriculados en la asignatura es superior a 400, divididos en grupos de 50 alumnos. Se dan dos sesiones de hora y media por semana, y una hora teórica para explicar conceptos generales de programación.

### **CONTENIDO:**

Laboratorio de Programación II: Se aprovecha de los conocimientos que se adquieren en Estructura de Datos y Algoritmos para realizar ejercicios con programación orientada a objetos. Se puede dividir la enseñanza de la asignatura en las siguientes partes:

### • Clase teórica.

Semanalmente se da una hora para explicar los conceptos generales y teóricos que sigue el lenguaje de programación y las técnicas y usos más frecuentes en programación orientada a objetos. Conseguimos de esta forma hacer llegar al alumno las bases teóricas que deben aplicar en las horas de prácticas.

Por regla general, la clase teórica adelanta los conceptos que van a ser tratado en futuras sesiones prácticas; intentamos siempre no adelantarnos demasiado en el contenido para que el alumno utilice los conocimientos que han adquirido en la cercana clase teórica.

Hacemos notar que la participación en la clase teórica no es muy alta, porque generalmente el alumno espera para comprobar los conceptos mencionados en las clases prácticas.

### • Sesiones prácticas de Borland Turbo Debugger 3.2.

Damos cuatro sesiones para el aprendizaje de esta herramienta que consideramos muy útil para un programador, sobre todo cuando estamos tratando con objetos y resulta algo engorroso hacer un seguimiento de la ejecución del programa.

Recordamos que una parte importante de la asignatura es la realización de una práctica usando PASCAL orientado a objetos; el uso del TD les va a resultar muy adecuado para encontrar fallos en la ejecución de forma dinámica que de otra manera serían difícilmente detectables.

Las primeras sesiones consisten en el aprendizaje de las opciones de la aplicación para, en sucesivas sesiones, ir incrementando la dificultad de los ejemplos para encontrar fallos de implementación.

### • Sesiones de explicación de la práctica de PASCAL con lenguaje orientado a objetos.

Una parte muy importante de la asignatura es el trabajo individual del alumno a la hora de enfrentarse al desarrollo de una aplicación con programación orientada a objetos. Para ello, los profesores proponemos un problema supuesto, que el alumno debe abordar siguiendo unos requisitos mínimos de implementación.

Los conceptos que el alumno debe dominar para la realización de la práctica son el uso clases con estructura jerarquizada, el uso de métodos virtuales, aplicación de la herencia, empleo de estructuras polimórficas y tipos abstractos de datos que se enseñan en la asignatura de Estructuras de Datos y Algoritmos. Se da por supuesto que el alumno domina las técnicas básicas de programación en PASCAL que aprendió en la asignatura de Laboratorio de Programación I.

Para que sirva de guía para la implementación, damos algunas clases sobre el contenido del enunciado de la práctica, debatiendo las estructuras de datos que se consideran más acertadas para cada problema y resolviendo las dudas que van surgiendo de las diferentes opciones.

El número de estas sesiones va a depender de cómo se vayan entendiendo todas las cuestiones que plantea el abordaje del diseño: de cualquier forma no suele ser superior a cuatro. Como es lógico, el mayor número de dificultades las vamos a resolver en horario de tutorías, cuando el alumno empiece a diseñar e implementar la práctica.

#### • Sesiones prácticas de programación en C y C++.

Suponiendo que el alumno no posee ninguna noción del lenguaje C, intentamos en estas sesiones que se asimilen las características en la programación en este lenguaje. Hacemos notar el inicial rechazo de los alumnos ante la falta de claridad del lenguaje C acostumbrados al didáctico PASCAL.

Mediante la explicación de las nociones básicas de cada concepto se le invita al alumno a que acometa la implementación de algunas prácticas breves para que asimile la nueva idea. Por supuesto la programación se hace de forma tutelada, y a menudo se van exponiendo en común problemas comunes y peculiaridades de la implementación.

Al contrario que en la clase teórica, la participación ahora es muy alta y entre los alumnos continuamente se intercambia información, lo que contribuye a la asimilación del conocimiento.

Como curiosidad, la parte que más cuesta ser asimilada es el uso de punteros, ya que se presenta como un concepto nuevo y de una visión no inmediata.

#### PROBLEMAS EXISTENTES:

El mayor problema que existe en las clases prácticas es la masificación del alumnado, que nos obliga a no tutelar como quisiéramos el aprendizaje de una forma más personalizada. Aunque no asisten a las prácticas la totalidad de los 50 alumnos en los que está dividido cada grupo, es opinión unánime entre profesores y alumnos la falta de seguimiento en el aprendizaje, aunque haya voluntad de todas las partes. Por lo general, y debido a la deficiencia de medios, suele haber dos personas por ordenador (alguna vez hasta tres), cuando lo deseable sería que cada práctica se resolviera individualmente.

Algunos de los ordenadores con los que cuenta la UEX para estas prácticas no son los más aconsejables por no renovarse paulatinamente el parque informático. Como muestra, en alguna de las salas de prácticas, los ordenadores no poseen ratón y se encuentran en monopuesto, limitando las posibilidades de uso.

Desde el punto de vista didáctico, se nos plantea el problema (en las sesiones de prácticas) de las diferentes capacidades de los alumnos que componen los grupos. Para cada sesión llevamos planteadas una serie de prácticas que el alumno debe resolver de forma consecutiva. Como debe ser la regla general no pasamos al siguiente ejercicio hasta no haber resuelto el anterior. La dificultad se nos presenta cuando existen alumnos que tienen gran capacidad en programación, con algún conocimiento previo o que practican en casa las lecciones que van aprendiendo, frente a los alumnos que se limitan a asistir a las clases y van resolviendo las prácticas. Intentamos solucionar este problema incitando a los alumnos más adelantados a que hagan variaciones sobre las prácticas propuestas y que investiguen en nuevas formas de implementación.