



UNIVERSIDAD DE LAS PALMAS
DE GRAN CANARIA

GUÍA DOCENTE

CURSO: 2020/21

40963 - TECNOLOGÍAS DE PROGRAMACIÓN

CENTRO: 180 - Escuela de Ingeniería Informática

TITULACIÓN: 4008 - Grado en Ingeniería Informática

ASIGNATURA: 40963 - TECNOLOGÍAS DE PROGRAMACIÓN

CÓDIGO UNESCO: **TIPO:** Obligatoria **CURSO:** 2 **SEMESTRE:** 1º semestre

CRÉDITOS ECTS: 6 **Especificar créditos de cada lengua:** **ESPAÑOL:** 6 **INGLÉS:**

SUMMARY

Programming Technologies is a course of the Programming subject, attached to the Software Development Engineering module, common to the branch of Computer Science Engineering. It is not, however, the student's first contact with programming.

This first contact occurs in the courses indicated as prerequisites. These courses provide the student with training in the imperative programming approach, using a general-purpose, imperative, structured, object-oriented, and weakly typed programming language, and also offer an introduction to a strongly typed language.

Programming Technologies delves into the subject using the object-oriented programming model as its axis, using a language that illustrates the main elements of this paradigm: classes, objects, inheritance, and polymorphism, and that also incorporates event-driven programming for GUI and concurrent programming.

REQUISITOS PREVIOS

- Fundamentos de Programación I
- Fundamentos de Programación II

Plan de Enseñanza (Plan de trabajo del profesorado)

Contribución de la asignatura al perfil profesional:

Tecnologías de Programación es una asignatura de la materia Programación, adscrita al módulo de Ingeniería de Desarrollo de Software, común a la rama de la Ingeniería Informática. No es, sin embargo, el primer contacto del estudiante con la programación.

Este primer contacto se produce en las asignaturas de primero indicadas como requisitos previos. Estas asignaturas proporcionan al alumno formación en el modelo de programación imperativa, utilizando un lenguaje de programación de propósito general, imperativo, estructurado, orientado a objetos y débilmente tipado. Además, ofrecen una introducción a un lenguaje fuertemente tipado.

Tecnologías de Programación profundiza en la materia usando como eje el modelo de programación orientado a objetos, empleando para ello un lenguaje que ilustra de forma sencilla los principales elementos de este paradigma: clases, objetos, herencia y polimorfismo y que también incorpora programación guiada por eventos y concurrencia.

Competencias que tiene asignadas:

CB1, CB5
G3, G4, G8, G9
N2, N3
CI1, CI8, CI11, CI14, CI17

Objetivos:

Que el alumno adquiera destreza en:

Ob1) Aplicar el paradigma de programación orientada a objetos en el desarrollo de programas.

Ob2) Utilizar técnicas avanzadas de programación orientada a objetos como uso y desarrollo de contenedores.

Ob3) Aplicar en el desarrollo de programas el modelo de programación guiada por eventos de forma adecuada y construir interfaces gráficas de usuario utilizando principios básicos de diseño.

Ob4) Diseñar y desarrollar aplicaciones concurrentes y distribuidas.

Contenidos:

1. Programación Orientada a Objetos

1.1. Introducción a la Programación Orientada a Objetos

1.2. Modularidad, herencia y polimorfismo

1.3. Genericidad y contenedores

Bibliografía: [1] [2] [3]

Competencias: CB1, CB5, G3, G4, G8, G9, N2, N3, CI1, CI8

2. Fundamentos de la Programación Guiada por Eventos e Interfaz Gráfica

2.1. Introducción, gestores de eventos y clases anidadas

2.2. Interfaz gráfica de usuarios

Bibliografía: [1] [2] [3]

Competencias: CB1, CB5, G3, G4, G8, G9, N2, N3, CI1, CI8, CI17

3. Programación Concurrente y Distribuida.

3.1. Introducción al paradigma de la programación concurrente.

3.2. Introducción a la programación distribuida.

Bibliografía: [1][2][4]

Competencias: CB1, CB5, G3, G4, G8, G9, N2, N3, CI1, CI8, CI11, CI14

CONTENIDOS PRÁCTICOS

Los contenidos prácticos de la asignatura consisten en la realización de 4 prácticas evaluables que faciliten al alumno adquirir las competencias relacionadas con los contenidos teóricos impartidos. En cada una de las prácticas el alumno desarrollará un programa o parte de un programa en que se empleen los contenidos mostrados en los temas teóricos relacionados.

Práctica 1. Clases, métodos y constructores. Herencia y polimorfismo (Tema 1).

Práctica 2. Contenedores (Tema 1).

Práctica 3. Interfaz gráfica de usuarios (Tema 2).

Práctica 4. Programación concurrente y distribuida (Tema 3).

La bibliografía necesaria para cada práctica coincide con la del tema teórico correspondiente. Las competencias a adquirir con la realización de cada práctica son las mismas que las del tema relacionado con la práctica.

Metodología:

La metodología a desarrollar incluirá las siguientes actividades formativas:

AF1) Sesiones académicas teóricas. El profesor expondrá y resolverá aspectos teóricos de los contenidos de la asignatura. Los contenidos se ilustrarán con ejemplos prácticos, preparando al estudiante para la realización de las actividades prácticas.

AF2) Sesiones académicas prácticas. El profesor expondrá las actividades prácticas a realizar, mostrando el uso de las herramientas necesarias para su realización y guiando en las dudas que se planteen en su desarrollo.

AF3) Ejercicios de autoevaluación. Se pondrá a disposición de los estudiantes ejercicios que permitan una autoevaluación sencilla de las habilidades adquiridas. Estas actividades se pueden realizar de forma no presencial.

AF4) Tutorías. Se usarán para resolver dudas y hacer un seguimiento de la asignatura.

PLAN DE CONTINGENCIAS NO PRESENCIAL

En caso de que la enseñanza de esta asignatura tuviera que pasar por causa de fuerza mayor a modalidad no presencial, se seguirá este mismo proyecto docente, sustituyendo las actividades presenciales por sus equivalentes telemáticos.

Evaluación:

Criterios de evaluación

En el proceso de evaluación de la asignatura se emplearán las siguientes fuentes de evaluación:

FE1) Exámenes síncronos individuales. Esta fuente de evaluación está relacionada con las actividades formativas AF1, AF2, AF3 y AF4.

FE2) Trabajos prácticos: individuales y en equipo. Esta fuente de evaluación está relacionada con las actividades formativas AF1, AF2, AF3 y AF4.

FE3) Coevaluación. Esta fuente de evaluación está relacionada con las actividades formativas AF2 y AF4.

FE1) Exámenes síncronos individuales

* Criterios relativos a las cuestiones:

-Conocimiento y comprensión de los contenidos

-Rigor y fluidez en la comunicación escrita

* Criterios relativos al desarrollo de un programa:

-El profesor someterá el programa a las pruebas oportunas para formular una valoración que tenga en cuenta todos los aspectos implicados en su realización, con especial énfasis en la ejecución (funcionamiento, adecuación a las especificaciones, robustez, ...) y en el estilo (formato, comentarios, elección de identificadores, ...).

FE2) Trabajos prácticos: individuales y en equipo

* Criterios relativos a una práctica:

-El profesor someterá los programas e información entregados a los análisis y pruebas oportunos para formular una valoración global que tenga en cuenta todos los aspectos implicados en su realización, con especial énfasis en la ejecución (funcionamiento, adecuación a las especificaciones, robustez, ...), en el estilo (formato, comentarios, elección de identificadores, ...) y en la autoría. Cuando lo estime conveniente, el profesor podrá citar a los alumnos para formularles cuestiones que considere relevantes para la valoración global.

FE3) Coevaluación

Cada miembro del equipo valora a sus compañeros teniendo en cuenta su grado de: colaboración, respeto, integración, distribución del trabajo, motivación, compromiso y conciliación. El profesor supervisará la apropiada y justa evaluación entre los miembros de los equipos.

FE4) Asistencia a sesiones académicas

En cada sesión académica se hará un control de asistencia.

Sistemas de evaluación

Los diferentes sistemas de evaluación de la asignatura se realizarán atendiendo a las distintas fuentes de evaluación:

* Evaluación de los exámenes individuales (SE1)

La escala que se utilizará para valorar esta parte es de 0 a 10 puntos y se calculará a partir de exámenes que se irán realizando regularmente para la evaluación continua y con el examen correspondiente en las convocatorias extraordinaria y especial. Estos exámenes se pueden basar en tests y/o cuestiones de respuesta corta y/o desarrollo de programas relativos a los temas tratados. Cada examen se valora de 0 a 10 puntos, aunque pondera en la nota SE1 según los temas evaluados.

En cada uno de los exámenes individuales de la evaluación continua, la nota se podrá reducir un 30% en caso de que el estudiante, injustificadamente, atienda a menos del 70% de las sesiones académicas teóricas correspondientes a los contenidos a examinar.

* Evaluación de los trabajos prácticos: individuales y en equipo (SE2)

La escala que se utilizará para valorar cada práctica es de 0 a 10 puntos. El enunciado de cada práctica incluirá las especificaciones, plazos de entrega y porcentaje que aporta a la evaluación; asimismo, se indicará el tipo de práctica: individual o en equipo. Se deberá entregar, dentro de los plazos establecidos, los ficheros conteniendo el código fuente y demás información requerida, usando el medio que se establezca. Para los trabajos en equipo, la nota obtenida en cada práctica es la que le corresponderá a cada alumno componente del equipo que la realiza. Estos trabajos solo se podrán realizar durante el periodo lectivo.

La nota de estos trabajos se podrá reducir un 50% en caso de que el estudiante no atienda como mínimo al 80% de las sesiones académicas prácticas.

* Coevaluación (SE3)

Cada alumno interviene en la evaluación al calificar la participación activa en el trabajo colaborativo de sus compañeros de equipo por medio de un cuestionario que se valorará de 0 a 10 puntos.

La nota de un alumno por cada coevaluación realizada será la media de la otorgada por sus compañeros de equipo. La nota final de esta parte será la media de los resultados obtenidos en las coevaluaciones efectuadas a lo largo del semestre y se valorará de 0 a 10 puntos. Requiere la atención a las sesiones académicas prácticas, con un mínimo del 80%. Los resultados de la coevaluación serán supervisados por el profesor para evitar posibles disfunciones en la nota.

PLAN DE CONTINGENCIAS NO PRESENCIAL

En caso de que la evaluación de esta asignatura tuviera que pasar por causa de fuerza mayor a modalidad no presencial, se seguirá este mismo proyecto docente, sustituyendo la evaluación presencial por sus equivalentes telemáticos.

Criterios de calificación

Para superar la asignatura en cualquier convocatoria es requisito obtener en SE1 una nota mínima de 5. En caso de no alcanzarse dicha nota se considera que las notas obtenidas en SE2 y SE3 valen cero.

El cálculo de la nota final (NF) se obtendrá aplicando la siguiente fórmula:

$$NF = SE1 * 0.7 + SE2 * 0.2 + SE3 * 0.1$$

La evaluación de la asignatura se realizará en la convocatoria ordinaria únicamente de forma continua.

La calificación en las convocatorias extraordinaria y especial se calculará como el máximo entre NF y SE1, sustituyendo SE1 en los cálculos por la obtenida en el examen de convocatoria.

Plan de Aprendizaje (Plan de trabajo de cada estudiante)

Tareas y actividades que realizará según distintos contextos profesionales (científico, profesional, institucional, social)

- Ta1) Búsqueda, estudio y consulta de información
- Ta2) Realización de ejercicios de programación autoevaluables.
- Ta3) Realización de prácticas de programación individuales
- Ta4) Realización de prácticas de programación en equipo que requieren del desarrollo individual para una posterior discusión colectiva.
- Ta5) Uso del entorno de desarrollo integrado para desarrollo de aplicaciones con interfaz gráfica y aplicaciones concurrentes

Científico: Ta1, Ta2, Ta3, Ta4, Ta5

Profesional: Ta1, Ta2, Ta3, Ta4, Ta5

Institucional:

Social: Ta1, Ta4

Temporalización semanal de tareas y actividades (distribución de tiempos en distintas actividades y en presencialidad - no presencialidad)

PRESENCIAL

Los estudiantes dedicarán 2 horas semanales a atender a sesiones académicas teóricas y otras 2 horas a sesiones académicas prácticas, que incluirán Ta2, Ta3, Ta4 y Ta5 (60 horas en 15 semanas). Los exámenes que se realizarán a lo largo del semestre consumen 7.5 horas en total.

NO PRESENCIAL

En promedio, realizarán cada semana 5 horas de trabajo autónomo: 2 horas dedicadas a la consulta y asimilación de materiales documentales (Ta1) y 3 horas para la realización de ejercicios y trabajos; ello supone 75 horas en 15 semanas.

Previsión de dedicación temporal en la relación de prácticas y actividades formativas:

- Semanas 1 a 3. Se realizarán Ta2 y Ta3.
- Semanas 4 a 7. Se realizarán Ta2 y Ta4.
- Semanas 8 a 11. Se realizarán Ta2 y Ta4 y Ta5.
- Semanas 12 a 15. Se realizarán Ta2, Ta4.

Las horas restantes las distribuirá el estudiante, en función de sus necesidades particulares, entre atención tutorial individual, tiempo adicional de estudio y realización de ejercicios y trabajos.

PLAN DE CONTINGENCIAS NO PRESENCIAL

En caso de que la enseñanza de esta asignatura tuviera que pasar por causa de fuerza mayor a modalidad no presencial, se seguirá este mismo proyecto docente, sustituyendo las actividades presenciales por sus equivalentes telemáticos.

Recursos que tendrá que utilizar adecuadamente en cada uno de los contextos profesionales.

Los recursos que utilizará el estudiante son los siguientes:

Re1) Navegador web

Re2) Moodle con VPL (Herramientas del campus virtual de la ULPGC, herramienta de programación online)

Re3) Herramientas de desarrollo integrado (Java, interfaz gráfica en Java)

CONTEXTOS

Científico: Re1, Re2, Re3

Profesional: Re1, Re2, Re3

Institucional:

Social: Re1

Resultados de aprendizaje que tendrá que alcanzar al finalizar las distintas tareas.

El estudiante deberá ser capaz de:

RA1) Desarrollar programas usando elementos avanzados de la programación orientada a objetos. Se adquiere con las actividades formativas AF1, AF2 y AF3.

RA2) Utilizar programación guiada por eventos para gestionar la interacción con los usuarios mediante interfaces gráficas. Se adquiere con las actividades formativas AF1 y AF2.

RA3) Construir interfaces gráficas de usuario utilizando componentes visuales comunes y principios básicos de diseño. Se adquiere con las actividades formativas AF1 y AF2.

RA4) Solucionar problemas mediante técnicas básicas de programación concurrente y programación distribuida. Se adquiere con las actividades formativas AF1 y AF2.

Alcanzar cada uno de estos resultados de aprendizaje puede requerir, según las dificultades que surjan durante su adquisición por parte de los alumnos, de la actividad formativa AF4.

Plan Tutorial

Atención presencial individualizada (incluir las acciones dirigidas a estudiantes en 5ª, 6ª y 7ª convocatoria)

La atención individualizada se verificará en el horario de tutorías establecido para los profesores de la asignatura y publicado en la web del Departamento de Informática y Sistemas.

No procede la atención individualizada de los estudiantes en 5ª, 6ª y 7ª convocatoria al ser una asignatura de nueva implantación.

Atención presencial a grupos de trabajo

La atención a grupos de trabajo se verificará en el horario de tutorías establecido para los profesores de la asignatura y publicado en la web del Departamento de Informática y Sistemas.

Atención telefónica

La atención telefónica se verificará en los teléfonos del profesorado publicados en la web del Departamento de Informática y Sistemas, en el horario de tutorías establecido para los profesores de la asignatura y publicado en la web del Departamento de Informática y Sistemas.

Atención virtual (on-line)

La atención virtual (on-line) se llevará a cabo usando las herramientas disponibles en el Campus Virtual de la ULPGC. Se responderá a las consultas virtuales en el horario de tutorías establecido para los profesores de la asignatura y publicado en la web del Departamento de Informática y Sistemas.

Datos identificativos del profesorado que la imparte.

Datos identificativos del profesorado que la imparte

D/Dña. José Daniel González Domínguez (COORDINADOR)
Departamento: 260 - *INFORMÁTICA Y SISTEMAS*
Ámbito: 570 - *Lenguajes Y Sistemas Informáticos*
Área: 570 - *Lenguajes Y Sistemas Informáticos*
Despacho: *INFORMÁTICA Y SISTEMAS*
Teléfono: 928458751 **Correo Electrónico:** *josedaniel.gonzalez@ulpgc.es*

Dr./Dra. David Sebastián Freire Obregón
Departamento: 260 - *INFORMÁTICA Y SISTEMAS*
Ámbito: 570 - *Lenguajes Y Sistemas Informáticos*
Área: 570 - *Lenguajes Y Sistemas Informáticos*
Despacho: *INFORMÁTICA Y SISTEMAS*
Teléfono: **Correo Electrónico:** *david.freire@ulpgc.es*

Dr./Dra. Juan Carlos Rodríguez Del Pino

Departamento: 260 - *INFORMÁTICA Y SISTEMAS*

Ámbito: 570 - *Lenguajes Y Sistemas Informáticos*

Área: 570 - *Lenguajes Y Sistemas Informáticos*

Despacho: *INFORMÁTICA Y SISTEMAS*

Teléfono: 928458733 **Correo Electrónico:** *jc.rodruiguezdelpino@ulpgc.es*

Bibliografía

[1 Básico] Piensa en Java /

Bruce Eckel.

Prentice Hall,, Madrid [etc.] : (2007) - (4ª ed.)

9788489660342

[2 Básico] The Java tutorials [

Oracle and /or its affiliates.

Oracle,, [s.l.] : (1995)

[3 Recomendado] Programador Java certificado: curso práctico /

Antonio J. Martín

Sierra.

Ra-Ma,, Paracuellos de Jarama, Madrid : (2010) - (3ª ed.)

978-84-7897-972-1

[4 Recomendado] Principles of concurrent and distributed programming /

M. Ben-Ari.

Pearson Education,, Harlow, England : (2006) - (2nd ed.)

9780321312839