



DATOS IDENTIFICATIVOS

Programación II

Materia	Programación II			
Código	O06G150V01205			
Titulación	Grao en Enxeñaría Informática			
Descritores	Creditos ECTS	Sinale	Curso	Cuadrimestre
	6	OB	1	2c
Lingua de impartición				
Departamento	Informática			
Coordinador/a	Garcia Perez-Schofield, Jose Baltasar			
Profesorado	Cuesta Morales, Pedro Garcia Perez-Schofield, Jose Baltasar Gonzalez Rufino, Maria Encarnacion			
Correo-e	jbgarcia@uvigo.es			
Web	http://fatic.uvigo.es			
Descrición xeral	Programación orientada a objetos.			

Competencias de titulación

Código	
A4	Coñecementos básicos sobre o uso e programación dos ordenadores, sistemas operativos, bases de datos e programas informáticos con aplicación na enxeñaría
A5	Coñecemento da estrutura, organización, funcionamento e interconexión dos sistemas informáticos, os fundamentos da súa programación, e a súa aplicación para a resolución de problemas propios da enxeñaría
A7	Capacidade para deseñar, desenvolver, seleccionar e avaliar aplicacións e sistemas informáticos, asegurando a súa fiabilidade, seguridade e calidade, conforme aos principios éticos e á lexislación e normativa vixente
A12	Coñecemento e aplicación dos procedementos algorítmicos básicos das tecnoloxías informáticas para deseñar solucións a problemas, analizando a idoneidade e complexidade dos algoritmos propostos
A13	Coñecemento, deseño e utilización de forma eficiente dos tipos e estruturas de datos máis axeitados á resolución dun problema
A14	Capacidade para analizar, deseñar, construír e manter aplicacións de forma robusta, segura e eficiente, elixindo o paradigma e as linguaxes de programación máis axeitadas
A25	Capacidade para desenvolver, manter e avaliar servizos e sistemas software que satisfagan todos os requisitos do usuario e se comporten de forma fiable e eficiente, sexan asequibles de desenvolver e manter e cumpran normas de calidade, aplicando as teorías, principios, métodos e prácticas da Enxeñaría do Software
A28	Capacidade de identificar e analizar problemas e deseñar, desenvolver, implementar, verificar e documentar solucións software sobre a base dun coñecemento axeitado das teorías, modelos e técnicas actuais
B1	Capacidade de análise, síntese e avaliación
B2	Capacidade de organización e planificación
B3	Comunicación oral e escrita na lingua nativa
B5	Capacidade de abstracción: capacidade de crear e utilizar modelos que reflectan situacións reais
B7	Capacidade de buscar, relacionar e estruturar información proveniente de diversas fontes e de integrar ideas e coñecementos
B8	Resolución de problemas
B9	Capacidade de tomar decisións
B10	Capacidade para argumentar e xustificar lóxicamente as decisións tomadas e as opinións
B11	Capacidade de actuar autonomamente
B12	Capacidade de traballar en situacións de falta de información e/ou baixo presión
B15	Capacidade de relación interpersoal
B16	Razoamento crítico
B18	Aprendizaxe autónoma
B19	Adaptación a novas situacións
B20	Creatividade

B22 Ter iniciativa e ser resolutivo

B24 Ter motivación pola calidade e a mellora continua

Competencias de materia

Resultados previstos na materia	Resultados de Formación e Aprendizaxe
Coñecer amplamente a linguaxe de programación orientada a obxectos de maior utilidade para a industria na actualidade.	A4 A5 A12 A13
Coñecer amplamente os procesos de análise e deseño asociados a un proceso de complexidade básica realizado mediante programación orientada a obxectos.	A7 A14 A25 A28
Desenvolver software de calidade aplicando os fundamentos do paradigma de orientación a obxectos.	A5 A12 A13 A14 A25 A28
Dominar a comunicación dentro do grupo de traballo e a exposición oral do traballo realizado.	B1 B2 B3 B5 B7 B8 B9 B10 B11 B12 B15 B16 B18 B19 B20 B22 B24

Contidos

Tema	
Introdución ao desenvolvemento orientado a obxectos.	Clases e obxectos. Punteiros. Xestión da memoria dinámica.
Fundamentos do modelo orientado a obxectos.	Sobrecarga de funcións e operadores. Heranza e composición. Excepcións Ligadura dinámica.
Genericidad. Xestión de excepcións. Ficheiros.	Plantillas. Librería STL. Ficheiros.

Planificación

	Horas na aula	Horas fóra da aula	Horas totais
Sesión maxistral	22.5	33.75	56.25
Resolución de problemas e/ou exercicios	28	37.8	65.8
Outras	0	22	22
Probas prácticas, de execución de tarefas reais e/ou simuladas.	2	4	6

*Os datos que aparecen na táboa de planificación son de carácter orientador, considerando a heteroxeneidade do alumnado.

Metodoloxía docente

Descrición

Sesión maxistral	Presencial: presentación, mediante medios audiovisuais, dos contidos teóricos de cada tema. Este método combinarase con exemplos ilustrativos de código e coa realización de preguntas para motivar e incrementar o interese do alumno.
	Non presencial: revisión, comprensión e afianzamento dos contidos.
Resolución de problemas e/ou exercicios	O obxectivo é que o alumno aplique os contidos teóricos na solución de problemas simples de programación.
	Presencial: resolución de pequenos problemas de programación.
	Non presencial: resolución de pequenos problemas de programación.

Atención personalizada

Metodoloxías	Descrición
Resolución de problemas e/ou exercicios	Realizarase un seguimento individualizado do alumno mediante titorías personalizadas e control do traballo semanal realizado.
Sesión maxistral	Realizarase un seguimento individualizado do alumno mediante titorías personalizadas e control do traballo semanal realizado.
Probas	Descrición
Probas prácticas, de execución de tarefas reais e/ou simuladas.	

Avaliación

	Descrición	Cualificación
Sesión maxistral	Cada tema terá unha proba individual, coa que se pretende comprobar si o alumno alcanzou os obxectivos de devandito tema. Existirán tres probas, unha por tema, valendo cada proba un 15% da nota final.	45
Resolución de problemas e/ou exercicios	Realizaranse tarefas continuas en grupo para a resolución de pequenos problemas de programación correspondentes a todos os temas de contidos da materia. Esta nota será proporcionada polo profesor de forma subxectiva.	25
Outras	Actividade de recuperación para os alumnos que non superen a materia na primeira opción.	0
Probas prácticas, de execución de tarefas reais e/ou simuladas.	Realizarase unha proba individual de programación, coa cal preténdese avaliar a capacidade do alumno ante o desenvolvemento de software de calidade.	30

Outros comentarios sobre a Avaliación

O proceso de avaliación para a segunda opción, e, para non asistentes, consiste en:

* unha proba individual por cada tema. Cada proba terá un valor do 15%, sendo en total o 45% da nota final.

* unha proba individual de programación, cuio valor corresponde ao 55% da nota final.

Tanto para alumnos que opten á primeira ou segunda opción, terase en conta que para aplicar as porcentaxes descritas é necesario que en calquera proba realizada obtéñase unha nota igual ou superior a 4, pero só se considerará superada a materia se a cualificación final é igual ou superior a 5.

Para os non asistentes que desexen presentarse en primeira opción, o día da proba individual de programación e da última proba individual por cada tema, realizarase ademáis un exame que reunirá as tres probas de mínimos.

Bibliografía. Fontes de información

L. Borrajo Diz, J. B. García Pérez-Schofield, **Diseño e realización de servicios e presentación en contornos gráficos**, Grupo Academia Postal,

E. Hernández Orallo, et al, **C++ estándar**, Paraninfo,

F. J. Ceballos, **Programación orientada a objetos con C++**, Ra-ma,

B. Stroustrup, **El Lenguaje de Programación C++**, Addison Wesley,

Harvey M. Deitel, Paul J. Deitel, **C++: cómo programar**, Pearson Educacion,

N. Dale, C. Weems, **Programación y resolución de problemas con C++**, Mc Graw-Hill,

E. Balagurusamy, **Programación orientada a objetos con C++**, Mc Graw-Hill,

L. Joyanes Aguilar, **Programación en C++: Algoritmos, estructuras de datos y objetos**, Mc Graw-Hill,

H. Schildt, **C++: guía de autoenseñanza**, Mc Graw-Hill,

C.H. Pappas, W. H. Murray, **C++ sin errores**, Mc Graw-Hill,

Al Stevens, **Programación con C++**, Anaya Multimedia,

J.P. Cohoon, **Programación y diseño en C++: introducción a la programación y al diseño orientado a objetos**, Mc Graw-Hill,

W. Savitch, **Resolución de problemas con C++**, Prentice Hall,

F. Llopis Pascual, **Introducción a la programación. Algoritmos y C/C++**, Publicaciones Universidad de Alicante,

F. J. Ceballos, **Enciclopedia del lenguaje C++**, Ra-ma,

Acera García, M. A., Sanz Sierra, A. M., **C/C++**. **Serie: Manuales Imprescindibles**, Anaya,

Liberty, J; Horvath, D. B., **Aprenda C++**. **Programación**, Anaya,

Responsable editorial, Víctor Manuel Ruiz Calderón, Susana Krahe Pérez-Rubín, **C++ estándar**. **Programación**, Anaya,

Alfonseca M., Sierra A., **Programación en C/C++**. **Guías Prácticas**, Anaya,

Eckel Bruce, **Thinking in C++**, On line,

Recomendaciones

Materias que se recomienda cursar simultaneamente

Informática: Algoritmos e estructuras de datos I/O06G150V01201

Materias que se recomienda ter cursado previamente

Informática: Programación I/O06G150V01104
