



Guía Docente				
Datos Identificativos				2023/24
Asignatura (*)	Concorrenza e Paralelismo	Código	614G01018	
Titulación	Grao en Enxeñaría Informática			
Descritores				
Ciclo	Período	Curso	Tipo	Créditos
Grao	2º cuatrimestre	Segundo	Obrigatoria	6
Idioma	CastelánGalegoInglés			
Modalidade docente	Presencial			
Prerrequisitos				
Departamento	Ciencias da Computación e Tecnoloxías da InformaciónComputaciónEnxeñaría de Computadores			
Coordinación	Paris Fernandez, Javier	Correo electrónico	javier.paris@udc.es	
Profesorado	Darriba López, Diego Fernández Fraga, Alejandro Fraguela Rodríguez, Basilio Bernardo González Domínguez, Jorge Paris Fernandez, Javier Quintela Carreira, Juan Jose Sanchez Penas, Juan Jose Tourinho Dominguez, Juan	Correo electrónico	diego.darriba@udc.es a.fernandez3@udc.es basilio.fraguela@udc.es jorge.gonzalezd@udc.es javier.paris@udc.es juan.quintela.carreira@udc.es juan.jose.sanchez.penas@udc.es juan.tourino@udc.es	
Web	campusvirtual.udc.es			
Descrición xeral				

Competencias do título	
Código	Competencias do título
A12	Coñecemento e aplicación dos procedementos algorítmicos básicos das tecnoloxías informáticas para deseñar solucións a problemas, analizando a idoneidade e a complexidade dos algoritmos propostos.
A20	Coñecemento e aplicación dos principios fundamentais e técnicas básicas da programación paralela, concorrente, distribuída e de tempo real.
B3	Capacidade de análise e síntese
C4	Desenvolverse para o exercicio dunha cidadanía aberta, culta, crítica, comprometida, democrática e solidaria, capaz de analizar a realidade, diagnosticar problemas, formular e implantar solucións baseadas no coñecemento e orientadas ao ben común.
C6	Valorar criticamente o coñecemento, a tecnoloxía e a información dispoñible para resolver os problemas cos que deben afrontarse.
C8	Valorar a importancia que ten a investigación, a innovación e o desenvolvemento tecnolóxico no avance socioeconómico e cultural da sociedade.

Resultados da aprendizaxe			
Resultados de aprendizaxe	Competencias do título		
	A12	B3	C4
Coñecemento e aplicación dos procedementos algorítmicos básicos das tecnoloxías informáticas para deseñar solucións a problemas, analizando a idoneidade e a complexidade dos algoritmos propostos dende o punto de vista da concorrenza e o paralelismo.			
Coñecemento e aplicación dos principios fundamentais e técnicas básicas da programación paralela, concorrente, distribuída e de tempo real.	A20		C6 C8

Contidos	
Temas	Subtemas



T1. Principios de programación concurrente	<ul style="list-style-type: none"> 1.1 Conceptos. <ul style="list-style-type: none"> 1.1.1 Arquitecturas hardware 1.1.2 Sistemas operativos 1.1.3 Procesos e Threads 1.2 Programación Multiproceso (fork/join) 1.3 Programación Multithread 1.4 Sección crítica. 1.5 Exclusión mutua 1.6 Instruccións atómicas 1.7 Sincronización mediante condicións 1.8 Semáforos <ul style="list-style-type: none"> 1.8.1 Mutex 1.8.2 Semáforos 1.9 Interbloqueo. Prevención, evitación, recuperación. 1.10 Inanición. 1.11 Comunicación e sincronización 1.12 Escalabilidade
T2. Algoritmos concurrentes	<ul style="list-style-type: none"> 2.1 Produtores/consumidores. 2.2 Lectores/escritores 2.3 Filósofos ceando 2.4 Compartición nula
T3. Principios de programación paralela	<ul style="list-style-type: none"> 3.1 Conceptos <ul style="list-style-type: none"> 3.1.1 Niveis de paralelismo 3.1.2 Dependencias de datos 3.2 Modelo de paso de mensaxes <ul style="list-style-type: none"> 3.2.1 Conceptos básicos 3.2.2 Comunicación punto a punto 3.2.3 Operacións colectivas 3.3 Análise de algoritmos paralelos <ul style="list-style-type: none"> 3.3.1 Medidas de prestacións de algoritmos paralelos 3.4 Metodoloxía da programación paralela <ul style="list-style-type: none"> 3.4.1 Descomposición en tarefas 3.4.2 Asignación de tarefas 3.4.3 Técnicas de optimización 3.5 Esquemas algorítmicos paralelos <ul style="list-style-type: none"> 3.5.1 Single Process Multiple Data 3.5.2 Paradigma Master/slave
T4. Deseño de algoritmos e aplicacións paralelas	<ul style="list-style-type: none"> 4.1 Bibliotecas de paso de mensaxes 4.2 Casos de estudo 4.3 Evaluación de rendemento 4.4 Aplicación de técnicas de optimización

Planificación				
Metodoloxías / probas	Competencias	Horas presenciais	Horas non presenciais / traballo autónomo	Horas totais
Sesión maxistral	A12 A20 C4 C6 C8	30	45	75
Proba mixta	A12 A20 B3 C4 C6	3	0	3
Prácticas de laboratorio	A12 A20 B3 C8	16	24	40
Solución de problemas	B3 C6	10	19	29



Proba práctica	A12 A20 B3	2	0	2
Atención personalizada		1	0	1

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

Metodoloxías	
Metodoloxías	Descrición
Sesión maxistral	Exposición oral complementada co uso de medios audiovisuais e a introdución de algunhas preguntas dirixidas ao alumnado, coa finalidade de transmitir coñecementos e facilitar a aprendizaxe.
Proba mixta	Exame da materia que combinará preguntas sobre a teoría con problemas a resolver.
Prácticas de laboratorio	Metodoloxía que fomenta o aprendizaxe a través da realización de actividades de carácter práctico, tales como desenvolvemento de exercicios de programación.
Solución de problemas	Resolución de problemáticas concretas xurdidas nas prácticas de laboratorio a teor dos coñecementos que se traballaron, os cales poden plantexar múltiples solucións.
Proba práctica	Probas de avaliación continua que trata dos contidos vistos nas prácticas e problemas ao longo do curso.

Atención personalizada	
Metodoloxías	Descrición
Prácticas de laboratorio Solución de problemas	Nas prácticas de laboratorio, seminarios e na resolución de problemas as persoas asistentes poderán presentar cuestións, dúbidas, etc. Atendendo a estas solicitudes, repasaranse conceptos, resolveranse novos problemas ou utilizarase calquera actividade que se considere axeitada para resolver as dúbidas.

Avaliación			
Metodoloxías	Competencias	Descrición	Cualificación
Proba práctica	A12 A20 B3	Probas de avaliación continua realizadas ao longo do curso sobre os temas traballados nas sesión de laboratorio e na docencia expositiva.	5
Prácticas de laboratorio	A12 A20 B3 C8	Realización de varios exercicios prácticos de concorrencia (bloque de prácticas de concorrencia) e unha serie de exercicios prácticos de paralelismo (bloque de prácticas de paralelismo). Inda que a realización das prácticas pode ser por parellas a súa avaliación é individual.	35
Proba mixta	A12 A20 B3 C4 C6	Proba sobre os contidos presentados na docencia expositiva e desenvolvidos nas sesións de laboratorio e nas titorías grupais. Estará dividida en dous bloques: concorrencia (temas 1 e 2) e paralelismo (temas 3 e 4), sendo cada bloque a metade da nota máxima da proba mixta.	60

Observacións avaliación



Para aprobar á asignatura é necesario obter unha cualificación global igual ou superior ao 50% aplicando as ponderacións da proba obxectiva, das prácticas de laboratorio e das probas prácticas.

O mecanismo de recuperación que se habilita é exclusivamente a repetición da proba mixta (60% da cualificación global da asignatura).

A cualificación de prácticas de laboratorio (35% da cualificación global da asignatura) e das probas prácticas obtida na convocatoria ordinaria consérvanse para a recuperación. Non é posible recuperar/mellorar para xullo a nota de prácticas de laboratorio ou das probas prácticas. A avaliación das prácticas de laboratorio farase obrigatoriamente no grupo asignado a cada alumno.

A

realización fraudulenta das probas ou actividades de avaliación, unha vez comprobada, implicará directamente a cualificación de suspenso na convocatoria en que se cometa: o/a estudante será cualificado con ?suspenso? (nota numérica 0) na convocatoria correspondente do curso académico, tanto se a comisión da falta se produce na primeira oportunidade como na segunda. Para isto, procederase a modificar a súa cualificación na acta de primeira oportunidade, se fose necesario.

Fontes de información

Bibliografía básica	<ul style="list-style-type: none">- Doug Lea (2000). Concurrent programming in Java design, principles and patterns . Reading, Massachusetts: Addison Wesley- Joe Armstrong (2007). Programming Erlang: Software for a Concurrent World. United States: Pragmatic Programmers- Francisco Almeida [et al.] (2008). Introducción a la Programación Paralela. Madrid: Paraninfo Cengage Learning- Peter S. Pacheco (1997). Parallel Programming with MPI. San Francisco, California : Morgan Kaufman
Bibliografía complementaria	- Wilkinson, B. y Allen, M.. (1999). Parallel Programming. Techniques and Applications Using Networked Workstations and Parallel Computers. . Upper Saddle River, New Jersey : Prentice Hall,

Recomendacións

Materias que se recomenda ter cursado previamente

Programación II/614G01006
Algoritmos/614G01011
Estrutura de Computadores/614G01012
Paradigmas de Programación/614G01014
Deseño Software/614G01015

Materias que se recomenda cursar simultaneamente

Sistemas Operativos/614G01016
Redes/614G01017
Proceso Software/614G01019

Materias que continúan o temario

Internet e sistemas distribuídos/614G01023

Observacións

(*A Guía docente é o documento onde se visualiza a proposta académica da UDC. Este documento é público e non se pode modificar, salvo casos excepcionais baixo a revisión do órgano competente dacordo coa normativa vixente que establece o proceso de elaboración de guías