



ARQUITECTURA DE COMPUTADORAS

1. Características generales

Nombre:	Arquitectura de Computadoras
Sigla:	CI-0120
Créditos:	4
Horas lectivas:	5 horas de teoría
Requisitos:	CI-0114 Fundamentos de Arquitectura, CI-0118 Lenguaje Ensamblador
Correquisitos:	Ninguno
Clasificación:	Curso propio
Ciclo carrera:	I ciclo, 3er año (énfasis de Ciencias de la Computación)
Docente:	Ileana Alpízar Arias
Dir-correo-e:	ileana.alpizar@ucr.ac.cr - ileana.alpizar@gmail.com
Asistente:	Por nombrarse
Grupo y semestre:	G1, II sem 2020
Horario de clase:	L 11 a 12:50 J 10 a 12:50
Horario de consulta:	L y J 13 a 15:30 y por correo electrónico.
Mediación:	Modalidad virtual alto , http://mediacionvirtual.ucr.ac.cr
Entorno:	II – S – 2020 – RRF – ARQUITECTURA DE COMPUTADORAS – 001

2. Descripción: Este curso está dirigido a estudiantes que inician su tercer año de bachillerato. Se enfoca en el estudio de distintas arquitecturas de computadoras para permitirle al estudiante tomar decisiones acertadas sobre las capacidades de un computador para ejecutar un determinado programa de manera eficiente, especialmente en ambientes de ejecución paralela. Se requieren conocimientos amplios de programación en C/C++, ensamblador y lenguaje de máquina. Durante el curso, el estudiante adquiere un mayor criterio que le permite evaluar y comparar el rendimiento de diferentes arquitecturas de computadoras. El curso puede darse bajo un enfoque por proyectos o mediante una combinación de clases magistrales, estudio independiente y tareas con el uso de laboratorios.

3. Objetivo General: Que cada estudiante comprenda los principios de diseño que rigen las principales arquitecturas de computadoras para determinar su impacto en el rendimiento de los programas.

4. Objetivos Específicos: Durante este curso el estudiante desarrollará habilidades para:

1. Relacionar la arquitectura de una computadora con su conjunto de instrucciones.
2. Utilizar técnicas que permitan obtener estadísticas dinámicas sobre los tipos de instrucciones ejecutadas.
3. Realizar y evaluar diferentes mediciones de desempeño de procesadores con el fin de comprender cómo y por qué la arquitectura del procesador en el que se trabaja es un elemento determinante en el comportamiento de las aplicaciones de software que ahí se ejecuten.
4. Analizar críticamente un microprocesador calculando medidas de rendimiento con base en sus elementos y la organización de su arquitectura.
5. Comparar y contrastar distintas organizaciones de memoria.
6. Aplicar técnicas que aprovechen el paralelismo inherente a un proceso.





7. Explicar las diferencias que existen entre distintos procesadores y multiprocesadores con implementaciones de paralelismo de hilos, de procesos y de datos.

5. Contenidos:

Objetivo específico	Eje Temático	Desglose
1 y 2	Fundamentos de diseño	El diseño de computadoras. Clases de computadoras. Definición de la arquitectura de una computadora. Tendencias en la tecnología de circuitos integrados y sus costos, consumo de energía y eficiencia. Medición del desempeño. Principios cuantitativos en el diseño de computadores.
1 y 3	Arquitectura basada en el conjunto de instrucciones	Clasificación. Direccionamiento de memoria. Modos de direccionamiento para procesamiento de señales. Tipo y tamaño de los operandos. Operandos para procesamiento de señales y multimedia. Operaciones para procesamiento de señales y multimedia. Instrucciones para el control de flujo. Codificación de un conjunto de instrucciones. Rol de los compiladores. Relación con lenguajes de alto nivel. Arquitectura MIPS (RISC-V).
1, 6 y 7	Paralelismo a nivel de instrucciones y ejecución segmentada (<i>pipeline</i>)	Definición de <i>pipelining</i> . Problemas con el pipeline: conflictos de datos y de control. Dificultades para implementar <i>pipelining</i> (excepciones). Pipeline para operaciones multiciclo y ejemplo del <i>pipeline</i> para el procesador MIPS (RISC-V). Procesadores superescalares (de emisión múltiple).
5	Jerarquías de memoria	Introducción a las memorias caché. <i>Translation lookaside buffers</i> (TLB). Tecnología de memoria y optimizaciones. Desempeño de la caché. Reducción de tasa de fallos de caché. Diseño de jerarquías de memoria y memoria virtual. Protección y ejemplos de memoria virtual.
4, 6 y 7	Paralelismo a nivel de datos	Arquitecturas vectoriales, SIMD (<i>single instruction, multiple data</i>) y GPU (<i>graphics processing unit</i>). Extensiones SIMD del conjunto de instrucciones para multimedios.
6 y 7	Paralelismo a nivel de hilos	Arquitecturas de memoria compartida centralizada. Multiprocesadores simétricos (SMP), Acceso uniforme a memoria (UMA). Desempeño de los multiprocesadores de memoria compartida simétrica. Coherencia de caché mediante protocolos <i>snooping</i> . Arquitecturas de memoria compartida distribuida no uniforme (NUMA). Coherencia de caché mediante protocolos de



		directorios.
5	Almacenamiento externo	Discos duros, memorias <i>flash</i> y discos de estado sólido (SSD).
3	Redes de interconexión y clusters	Creación de procesos en ambientes de multiprocesadores.

6. Metodología:

Debido a que se desarrollará un **proyecto programado** durante el curso, los temas del curso girarán en torno a éste conforme se vayan necesitando.

Como la UCR se encuentra ofreciendo las **clases en modo virtual** debido a la pandemia del SARS-Cov 2, nuestro curso se dará de manera combinada entre clases sincrónicas utilizando ZOOM y actividades asincrónicas principalmente videos, los cuales, sin embargo, pueden verse durante las horas de clase asignadas al curso, ya que las clases sincrónicas se darán solo en uno de los dos días de clase.

Todas las **clases sincrónicas serán grabadas** y el video se pondrá a disposición de los estudiantes, para referencia posterior.

7. Evaluación:

1er Examen (J 1° de Octubre)	25 %
EXAMEN FINAL (Acumulativo) (M 2 de Diciembre)	25 %
PROYECTO PROGRAMADO	32 %
Tareas cortas	18 %

Las fechas y horas anteriores podrían cambiar. Reposición de exámenes solo en casos realmente justificados

8. Cronograma tentativo

TEMA	I	II	III	IV	V	VI	VII	VIII
SEMANAS	1 y 2	3, 4, 5 y 6		7, 8, 9 y 10		11, 12 y 13	14	15

9. Bibliografía:

[1] JOHN HENNESSY & DAVID PATTERSON, “COMPUTER ARCHITECTURE A QUANTITATIVE APPROACH”, VI Edition. Morgan Kaufmann – Elsevier Inc. Publishers, 2019 (**Libro de texto**)

[2] Se utilizarán varias **lecturas de artículos** que serán escogidos durante el semestre de acuerdo con el proyecto y su temática.



10. Aspectos relacionados con el sistema de bibliotecas, acceso a los reglamentos estudiantiles, y evaluación por parte de los profesores

El Sistema de Bibliotecas, Documentación e Información (SIBDI) de la Universidad de Costa Rica

(<http://sibdi.ucr.ac.cr/>) cuenta con una amplia gama de recursos de información bibliográfica en diferentes formatos como libros, folletos, publicaciones periódicas, trabajos finales de graduación, entre otros. Algunos de estos recursos se encuentran en Biblioteca Virtual, desde la cual se pueden acceder las publicaciones de conferencias y revistas de ACM, IEEE (<http://sibdi.ucr.ac.cr/dbingenieria.jsp>), o Springer, entre otras. La Biblioteca Luis Demetrio Tinoco ofrece cursos de capacitación a los estudiantes del área de las Ingenierías y Computación.

El sitio web del Consejo Universitario de la UCR contiene las diferentes normativas estudiantiles, que pueden ser consultadas desde el enlace <http://www.cu.ucr.ac.cr/normativa/estudiantil.html>.

Los procedimientos de evaluación y orientación establecidos en el Reglamento de Régimen Académico Estudiantil se encuentran en http://www.cu.ucr.ac.cr/uploads/tx_ucruniversitycouncildatabases/normative/regimen_academico_estudiantil.pdf

Destacamos especialmente los siguientes artículos de dicho Reglamento:

- El Artículo 14 se refiere al contenido que debe tener un programa del curso, incluyendo “las normas de evaluación desglosadas y con las ponderaciones de cada aspecto a evaluar”.
- El Artículo 17 indica que “las normas de evaluación conocidas por los estudiantes pueden ser variadas por el profesor con el consentimiento de la mayoría absoluta (más del 50% de los votos) de los estudiantes matriculados en el curso y grupo respectivo”.
- El Artículo 22 indica que “el profesor debe entregar a los alumnos las evaluaciones calificadas y todo documento o material sujeto a evaluación, a más tardar diez días hábiles después de haberse efectuado las evaluaciones y haber recibido los documentos”.
- El Artículo 24 establece el procedimiento y fechas para realizar la reposición de evaluaciones.