

PLANIFICACIÓN DE LA ASIGNATURA
ARQUITECTURA DEL COMPUTADOR

1.- IDENTIFICACIÓN

1.1- **ASIGNATURA:** Arquitectura del Computador

1.2- **CARRERA:**
PROFESORADO EN INFORMATICA

1.3- Ubicación de la Asignatura en el Plan de Estudios

1.3.1- 4º Módulo
2º Año

La asignatura pertenece al **Primer Ciclo.**

1.3.2- Área

Arquitectura, Sistemas Operativos y Redes.

1.3.3- Carga horaria: **75 hs.**

1.3.4- Cuadro sintético de la carga horaria

Horas teóricas	Horas de Formación practica					Total
	Formación experimental	Resolución de problemas del mundo real	Actividades de proyectos y diseño de sistemas de información	Instancias supervisadas de formación en la práctica profesional	Otras Practicas?	
30	15				30	75

1.3.5- Correlativas Anteriores:

- Organización del Computador (regular)
- Lógica II (regular)
- Ingles I (aprobada)

1.3.6- Correlativas Posteriores:

- Sistemas Operativos
- Redes

1.4- Objetivos establecidos en el Plan de Estudios para la Asignatura.

El Plan de estudios no tiene definidos objetivos por asignatura

1.5- Contenidos mínimos establecidos en el Plan de Estudios para la Asignatura

Clasificación de las arquitecturas de computadoras. Máquina de Von Neumann. Microarquitectura. Trayectoria de datos. Ciclos de búsqueda y ejecución. Decodificación de instrucciones. Las líneas de desarrollo de las arquitecturas Von Neumann. Jerarquía de memoria. Memorias caché. Pipelining. Lenguaje ensamblador. Arquitecturas no Von Neumann. Arquitecturas multiprocesadores. Máquinas Algorítmicas. Procesadores de alta prestación. Arquitecturas reconfigurables. Arquitectura orientada a servicios. Arquitecturas Grid.

1.6- Carga horaria semanal y total:

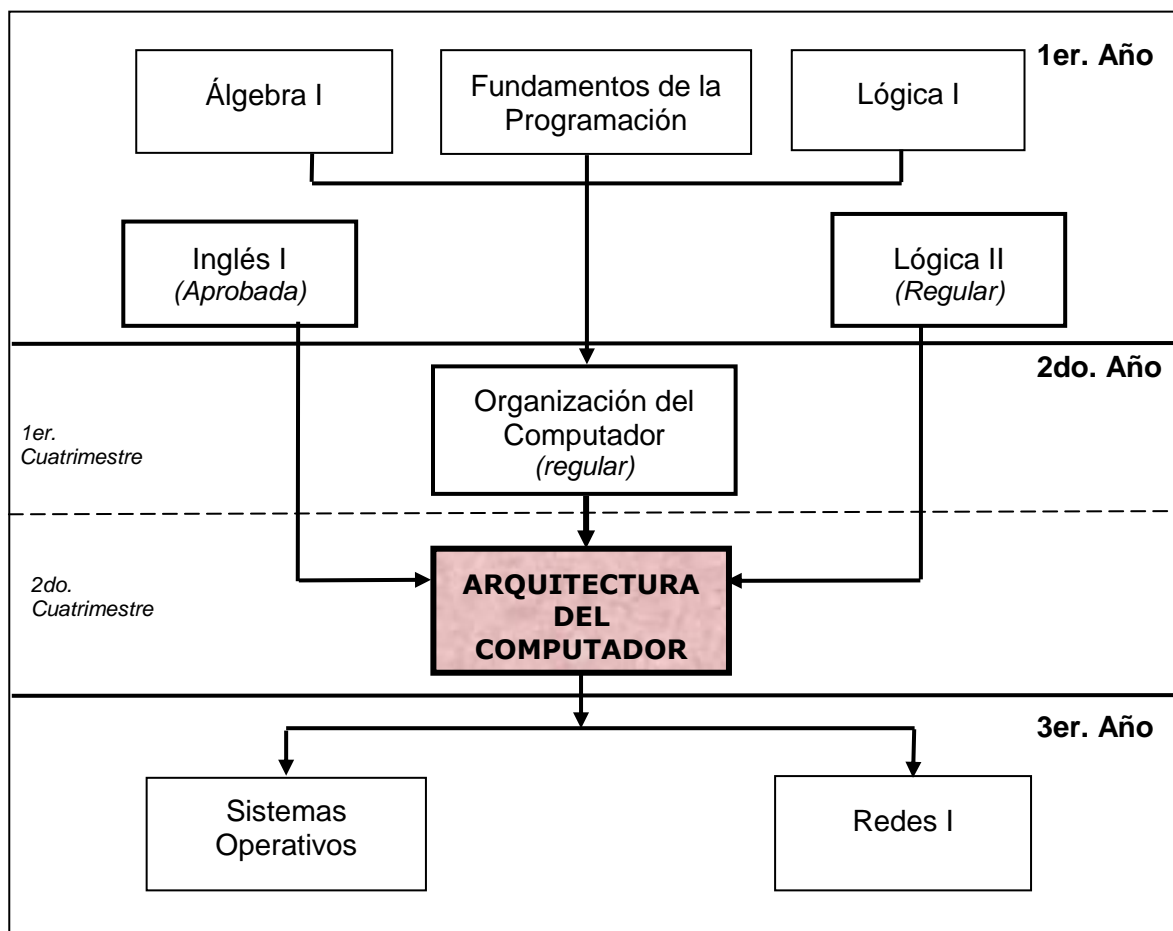
5 horas semanales
75 horas totales.

1.7- Año académico: 2013

2.- PRESENTACIÓN

2.1- Ubicación de la Asignatura como tramo de conocimiento de una disciplina.

Ubicación de la asignatura en el plan de estudios



2.2- Conocimientos y habilidades previas

El alumno deberá tener una idea general del hardware de un procesador de propósito general. su estructura lógica y su organización. La estructura a nivel de sistema de compuertas lógicas, circuitos lógicos y transporte de señales lógicas, y su organización para interpretar un lenguaje reconociendo y ejecutando órdenes.

Requiere además conocimiento de los sistemas de numeración, el álgebra de Boole, y la lógica de las funciones construidas con elementos binarios.

Debe estar entrenado para entender algoritmos sencillos y expresarlos en algún código formalizado.

3.- OBJETIVOS

3.1- Objetivos Generales.

- Que el alumno adquiera habilidad para:
 - Analizar sistemas computacionales complejos desde el enfoque sistémico.
 - Inferir aspectos relacionados al funcionamiento interno de un microcomputador a partir de observaciones.
 - Justificar la dinámica del comportamiento de un computador autorregulado.
 - Pensar holísticamente relacionando el todo (sistema computacional) con las partes (CPU, memoria, E/S, etc.) y viceversa.
- Que el alumno desarrolle:
 - Una actitud de apertura hacia el avance tecnológico.
 - Autogestión de su aprendizaje.
 - Responsabilidad, respeto y compromiso con el trabajo propio y de sus pares.

3.2- Objetivos Específicos.

- Que el alumno logre:
 - Internalizar conceptos relacionados a la estructura interna del computador.
 - Comprender la organización de los distintos componentes de un procesador y sus interrelaciones.
 - Identificar los distintos niveles funcionales de una computadora autorregulada.
 - Utilizar lenguajes de máquina de nivel convencional
 - Diferenciar arquitecturas no convencionales.

4.- SELECCIÓN Y ORGANIZACIÓN DE CONTENIDOS

4.1- Programa Sintético sobre la base de los Contenidos Mínimos

Clasificación de arquitecturas de computadoras. Lenguaje ensamblador. Máquinas algorítmicas. Procesadores de alta prestación. Arquitecturas no Von Newman. Procesamiento paralelo y procesamiento por flujo de datos. Conceptos de Arquitecturas Grid, arquitecturas reconfigurables y arquitecturas orientadas a servicios

4.2- Programa Analítico

Unidad I: ARQUITECTURA Y ORGANIZACIÓN DE COMPUTADORAS

1. Sistema basado en microprocesador.
 - Máquinas que ejecutan instrucciones. Ejemplos en procesadores típicos.
 - Análisis del conjunto de instrucciones de procesadores de uso comercial.
 - Ejecución solapada ("pipeline"). Aplicación en procesadores actuales.
 - Análisis de prestaciones.
 - Arquitecturas reconfigurables: conceptos.
2. Nivel de Microarquitectura
 - Unidad de control
 - Memoria de control
 - Microprograma
 - Ciclo de búsqueda y ejecución de instrucciones mediante microprograma
 - Cronología de microinstrucciones
 - Secuenciamiento de microinstrucciones
3. Ejemplo Arquitectura Microprogramada de maquina actual (Pico Java).

Unidad II: SUBSISTEMA DE ENTRADA/SALIDA

1. Subsistemas de entrada/salida.
 - E/S y su relación con la CPU, tipos de puertas.
 - Puerta de Entrada y Salida paralelo. Puerta de Entrada y Salida serie.
 - Tipos de transmisión serie.
 - Formato de transmisión serie asincrónica y sincrónica.
 - Entrada/Salida mapeada en memoria y con espacio dedicado de direccionamiento.
2. Mecanismos de Atención de las Entradas y Salidas

- Administración de las puertas por encuesta (polling) o por interrupción.
- Tratamiento de la CPU de las operaciones de E/S, por interrupción o por software.
- Transferencias de E/S por hardware, DMA, implementación.

Unidad III: MEMORIA

1. Características de los sistemas de memoria
 - Jerarquía de memoria
2. Mejoras de desempeño
 - Memorias caché, interleaving.
 - Cache: arquitectura; direccionamiento; organización; manejo de lecturas y escrituras; coherencia.
 - Acceso directo a memoria: Descripción del mecanismo. Controladores para DMA.
3. Microprocesador Intel 8086/8088/Pentium
 - Ejemplo de Memoria segmentada
 - Ejemplo de Formatos de Instrucción
 - Ejemplo de Modos de direccionamiento
 - Ejemplo de interrupciones
 - Pipeline: conceptos básicos

Unidad IV: ARQUITECTURAS NO CONVENCIONALES

1. Arquitecturas RISC
 - Características.
 - Principios de diseño de RISC
 - RISC versus CISC
 - Ejemplos de arquitectura RISC
2. Arquitecturas en paralelo.
 - Descripción general
 - Multiprocesadores
 - Máquina de conexión
 - Computadora de flujo de datos
3. Procesadores Pentium, IA-32, IA-64: generalidades
4. Conceptos introductorios: Maquinas algorítmicas, Arquitecturas Orientadas a Servicios (SOA) y Arquitecturas GRID

4.4- Programa y cronograma de Trabajos Prácticos

Trabajo práctico N° 1: Microprogramación

Trabajo práctico N° 2: Memoria caché

Trabajo práctico N° 3: Modos de direccionamiento

Trabajo práctico N° 4: Interrupciones

Trabajo práctico N° 5: Arquitecturas No convencionales

4.5- Programa y cronograma de Laboratorio

Taller 1: Maquina Virtual Pico Java

Taller 2: Microprocesador 8088: Direccionamiento e interrupción

5- BIBLIOGRAFÍA

5.1- Bibliografía General

- Patterson, D. A.; Hennessy J L.
Estructura y diseño de computadores. Interficie circuitería/programación
Editorial Reverté, S.A., 2000.
- Angulo Usategui J. y otros
Arquitectura de Microprocesadores. Los Pentium a fondo
Editorial Paraninfo, 2004
- Hamacher C. y otros
Organización de Computadores
Editorial Mcgraw-Hill, 5ta Edición, 2003.
- Tocci, R.; Widmer, N.
Sistemas Digitales: principios y aplicaciones
Editorial Pearson Education, México, 2003.

5.2- Bibliografía Específica

- Stalling William
Organización y arquitectura de computadores
Prentice Hall, 5ta. Edición (2001) y 7ma. Edición (2006) Madrid- España
- Tanenbaum Andrew S.
Organización de Computadoras: un enfoque estructurado
Prentice Hall, 4ta. Edición (2003) y 5ta Edición (2006), Madrid- España

6.- ESTRATEGIAS METODOLÓGICAS

6.1- Aspectos pedagógicos y didácticos

Bajo el supuesto que didácticamente no puede haber una distancia en sentido excluyente entre teoría y práctica se asume que existe un predominio de uno a otro aspecto dando lugar a lo que denominamos clases teóricas y clases prácticas.

En las clases teóricas la exposición explicativa va auxiliada del uso intensivo del pizarrón, donde utilizan como recursos didácticos lenguajes gráficos (tablas, esquemas, gráficos, etc.). La presentación de cada tema va acompañada de algún ejercicios de aplicación representativo que permita el diálogo y discusión para reforzar conceptos, aclarar dudas, reiterar explicaciones, escuchar propuestas etc.. Este estadio se complementa con la práctica y los talleres, y constituyen tres espacios nutren mutuamente.

Las clases prácticas se desarrollan utilizando como recurso didáctico las guías elaboradas especialmente para profundizar y afianzar los conceptos teóricos desarrollados, y en consecuencia tender a que el alumno logre autogestión en su aprendizaje.

En esta asignatura se implementan dos talleres en laboratorio como estrategia holística para integrar lo conceptual y lo práctico y de esta manera lograr que los alumnos logren una comprensión total de la arquitectura interna de una computadora, su estructura y organización.

6.2- Actividades de los Alumnos y de los Docentes

Actividades del equipo cátedra

La asignatura está a cargo de un equipo docente conformado por un profesor adjunto, el jefe de trabajos prácticos y el ayudante estudiantil. Los docentes desempeñan las siguientes funciones:

- Facilitador del aprendizaje mediante recursos didácticos específicos,
- Observador del proceso de aprendizaje,
- Promotor de la comunicación entre pares y docentes – alumnos,
- Animador del trabajo grupal para el tratamiento de los contenidos,
- Consultor permanente frente a las dudas e inquietudes de los alumnos.

El profesor adjunto es el responsable del desarrollo general de la asignatura. En particular se encarga de la coordinación del equipo cátedra y de la elaboración de la planificación de la asignatura. Se encarga del desarrollo de las clases teóricas, atención de consultas y del seguimiento y evaluación permanente de los alumnos.

El jefe de trabajos prácticos se encarga de las clases prácticas, elaboración de trabajos prácticos y taller.

El Ayudante Estudiantil colabora en la preparación de los trabajos prácticos y material didáctico, en el desarrollo de los talleres y atención de consultas de los alumnos.

El equipo cátedra acuerda cuestiones relativas al desarrollo de la asignatura, definición de roles y funciones aspectos inherentes a la implementación de las clases (selección de material bibliográfico, elaboración del material didáctico, preparación de las evaluaciones y planificación y desarrollo del taller)

Actividades de los alumnos

Los alumnos se encargan de realizar las actividades siguientes:

- Participar de las discusiones sobre los temas que se traten en cada clase.
- Resolver trabajos prácticos individualmente y/o en grupo según los requerimientos del docente.
- Evaluar la propia práctica en función a una puesta en común guiada por el docente.

6.3- Cuadro sintético

Tabla 1. Distribución de la carga horaria entre clases teóricas, práctica y taller

Clase	Carga Horaria	A cargo de	Técnica más usada	Énfasis en
Teórica	2/semanas	Prof. Adjunto Ing. M. Sosa Auxiliar de 1ra. Lic. I. Maldonado	Clases expositivas, reforzadas con ejemplos prácticos	Conceptualización formal.
Práctica	3/semanas	Prof. Adjunto Ing. M. Sosa Auxiliar de 1ra. Lic. I. Maldonado	Desarrollo de guías de trabajos prácticos Aplicación de enfoques que estimulen la creatividad para el diseño y desarrollo de soluciones	Resolución de problemas
Taller en laboratorio	15 hs. distribuidas en el cuatrimestre	Prof. Adjunto Ing. M. Sosa Auxiliar de 1ra. Lic. I. Maldonado Un Auxiliar de 2da.	Utilización de Software de tipo educativos y uso de debuggers La observación detenida y reflexiva de las propuestas.	Formulación y resolución de problemas Reafirmación de conceptos

- N° de alumnos estimado: la asignatura se organiza en dos comisiones donde cada una cuenta con 30 alumnos aproximadamente.

- Asistencia exigida:
 - Clases teóricas: 80%
 - Clases prácticas: 80%
 - Taller: 100%
 - Evaluativos: 100%

6.4- Recursos Didácticos

- Pizarrón: se lo utiliza principalmente como soporte a las clases expositivas.
- Transparencias: se las utiliza cuando se las considera pertinente para enriquecer las presentaciones de los temas durante las clases expositivas.
- Computadoras de tipo PC: utilizadas para la implementación del software usado para que el estudiante logre conceptualizar los elementos del microprocesador estudiado y le ayuden a formar una imagen clara y exhaustiva de su funcionamiento.
- Software *debugger* del procesador de Intel 8088, utilizada para estudiar la estructura, direccionamiento, interrupciones y lenguaje de assembler.
- Software para maquina virtual Pico Java, para estudiar los aspectos de la microprogramación.

7.- EVALUACIÓN

7.1- Evaluación Diagnóstica

No se realiza

7.2- Evaluación Formativa

Se realiza en forma continua durante las clases teóricas y clases prácticas. A medida que se introducen nuevos conocimientos se proponen ejercicios a realizar. Esos ejercicios se controlan y registran en planillas para que posteriormente permitan hacer recomendaciones metodológicas para el aprendizaje y seguimiento de la materia.

7.3- Evaluación Parcial

La evaluación parcial es diseñada con ejercicios prácticos correspondientes a una o varias unidades a evaluar y se la realiza por escrito. Se evalúa diferentes aspectos relacionados al procedimiento metodológico y los recursos utilizados para dar respuesta a un problema planteado, y la precisión de la respuesta.

Las escalas de evaluación son:

- (1-4) desaprobado,
- (5-6) suficiente,
- (7-10) promocionado.

Tabla 2. Distribución temporal de evaluaciones parciales y talleres

Evaluación	Contenidos	Fecha probable
Parcial I	Unidad 1 y parte de la 2 y 3	Semana 8
Parcial II	Unidad 4 y parte de la 3	Semana 13
Recuperatorio	Integral o por parcial según corresponda	Semana 15
Taller 1	Unidad 1	Semana 7
Taller 2	Unidad 3	Semana 12

7.4- Evaluación Integradora

La evaluación integradora se realiza sobre el desarrollo de un trabajo de taller. Este trabajo se defiende en un coloquio no eliminatorio, que se realiza para evaluar el conocimiento total adquirido.

En esta instancia se definen los siguientes criterios de evaluación:

- *Manejo de vocabulario técnico,*
- *Capacidad de resolución de problemas planteados,*
- *Utilización de recursos funcionales del microcomputador.*

Esta evaluación se enfatiza principalmente en los estudiantes candidatos a promocionar la asignatura para reunir todos los antecedentes de las evaluaciones anteriores definir la nota de promoción final.

7.5- Autoevaluación

Se motiva a los estudiantes a hacer un juicio crítico sobre el desarrollo de la asignatura, y el desempeño de los docentes y ayudantes. Las opiniones, sugerencias y críticas de los alumnos sirven como retroalimentación de la cátedra con el objeto de mejorar la propuesta de enseñanza.

7.6- Evaluación Sumativa

Para realizar la evaluación sumativa se reúnen los antecedentes de participación en las clases teóricas y prácticas, evaluaciones parciales y coloquio.

Para la promoción se debe reunir los porcentajes de asistencia establecidos, aprobar los parciales con un mínimo de 7 puntos y aprobación de los talleres.

Para la Regularidad se debe reunir los porcentajes de asistencia establecidos, aprobar los parciales o los recuperatorios correspondientes, con un mínimo de 5 puntos y aprobación de los talleres.

7.7- Examen final

La evaluación final será escrita u oral sobre los temas incluidos en la programación analítica de la asignatura.

Los aspectos que se evalúan son:

- Conocimientos de los aspectos estructurales y funcionales de los sistemas de computación estudiados a nivel de microarquitectura y nivel de máquina convencional.
- Utilización de los lenguajes formales y herramientas de diseño de componentes estructurales y de software.

7.8.- Examen Libre

El examen libre se divide en dos etapas, práctica y teórica, cada una de ellas es eliminatoria:

1)- Práctica: evaluación escrita de tipo práctica y desarrollo de un trabajo equivalente al del taller que realizan los alumnos regulares. El trabajo de taller debe ser aprobado por lo menos con 2 días de anticipación al día del examen final. Es decir el alumno requerirá el enunciado del trabajo 10 días antes del examen final.

2)- Teoría: la evaluación será en forma oral.

Ing. Mabel Sosa
Docente responsable de la asignatura