

## PRONTUARIO

<b>TÍTULO:</b>	Arquitectura de computadoras
<b>CODIFICACIÓN:</b>	CCO 235
<b>PRERREQUISITO:</b>	CCO 130
<b>CRÉDITOS:</b>	4 créditos   4 horas semanales   1 semestre

## DESCRIPCIÓN

Curso teórico y práctico fundamental sobre la arquitectura de las computadoras, sus estructuras internas y el procesamiento de datos digitales en el nivel más bajo posible, justo antes de ser convertidos en señales eléctricas. Se estudian los conceptos de lógica digital, las compuertas más primitivas y cómo son utilizados para construir circuitos lógicos. Los estudiantes practican lo aprendido creando circuitos digitales que resuelvan problemas reales. Se estudia la codificación de programas de bajo nivel. Los estudiantes crean varios programas en lenguaje de ensamblador para resolver una diversidad de problemas matemáticos y lógicos.

## JUSTIFICACIÓN

La arquitectura del computador es la estructura que ve un programador al emplear un lenguaje de ensamblador. Esa misma arquitectura es la que procesa y ejecuta todos los programas que sean creados sin importar el lenguaje de alto nivel que haya sido utilizado. El conocimiento de ésta estructura permite al programador seleccionar estrategias más eficiente para desarrollar sistemas automatizados de información. Los estudiantes de informática que deseen prosperar en el desarrollo eficiente y avanzado de sistemas de tecnología deberán tener un dominio pleno del funcionamiento interno de los equipos, representación de los datos y la creación de algoritmos lógicos para circuitos primitivos. El buen entendimiento de las soluciones programáticas que resuelvan problemas del mundo real requiere el conocimiento de cómo funcionan los circuitos electrónicos y cómo se procesan los datos en lenguaje de bajo nivel que realmente los ejecuta.

## COMPETENCIAS

El curso desarrolla en el o la estudiante las siguientes competencias:

### Competencias transversales

1. Razonamiento matemático (Nivel III)
2. Cuestionamiento crítico (Nivel III)

3. Solución de problemas (Nivel III)
4. Creatividad (Nivel II)
5. Tecnología de Información y Comunicación (Nivel III)

### **Competencias profesionales**

1. Actitud de apreciar la profundidad intelectual, los fundamentos abstractos, los planteamientos filosóficos y los valores estéticos que juegan un papel importante en el desarrollo de la disciplina y que continuamente presentarán retos a investigadores en el futuro.
2. Habilidad para describir un problema con claridad y estudiar, especificar, diseñar, implantar, probar, modificar y documentar la solución al problema.

### **OBJETIVOS**

Al finalizar el curso el o la estudiante será capaz de:

1. Demostrar habilidad de razonamiento matemático para representar una variedad de números complejos, datos y estructuras análogas utilizando estrictamente el sistema binario.
2. Demostrar habilidad para reconocer los dispositivos lógicos más básicos, distinguir las distintas compuertas lógicas existentes y como se combinan para crear los circuitos secuenciales comúnmente utilizados para resolver creativamente problemas cotidianos.
3. Demostrar conocimientos sobre la organización interna de un computador así como la función de cada una de sus partes en el contexto del procesamiento lógico de los datos.
4. Demostrar habilidad para aplicar los conceptos de solución de problemas lógicos para crear un algoritmo y desarrollar un programa de bajo nivel que resuelva un requerimiento en particular.
5. Demostrar conocimientos históricos sobre las personas importantes en el desarrollo de las tecnologías de circuitos y programación de bajo nivel utilizadas hoy día.

### **CONTENIDO**

- I. Introducción: Conceptos Generales
  - A. Definición de términos
  - B. Características Computadoras
  - C. Historial de CPUs
  - D. La máquina Universal de Turing
  - E. Solución de Problemas utilizando computación

1. Problema
  2. Algoritmo
  3. Modularización
  4. Abstracción
- F. Investigar sobre personajes que influenciaron la tecnología de cómputos
1. Alan Turing
  2. Dennis Ritchie
  3. Ken Thompson
  4. Gordon Moore
- II. Bits, Bytes y Palabras
- A. Representación de datos en las computadoras
- B. Sistema binario avanzado y repaso de conversiones
- C. Representación binaria de data tipos
1. Números
  2. Textos
- D. Manejo de números
1. Enteros sin signo
  2. Enteros con signo
  3. "Two's Complements"
- E. Operaciones Aritméticas
1. Suma, resta y "carry forward"
  2. Fracciones, decimales pequeños y decimales grandes
- F. Más allá de los números
1. Operaciones Lógicas y Booleanas
  2. "CodePages" y "Character Sets"
  3. ASCII y Unicode (ISO 8859-1)
  4. Imágenes, gráficas, "bitmaps"
  5. Sonidos, "sampling", "sampling rate"
- III. Estructuras Lógicas Digitales
- A. Compuertas Lógicas
1. Circuito simple: interruptor
  2. Transistores MOS
    - a. n-type
    - b. p-type

- 3. Logic Gates and CMOS
  - a. and, or, not, nor, nand
- 4. Ley DeMorgan's Law
- 5. Circuitos Combinados y Secuenciales
  - a. Decoder
  - b. Multiplexer
  - c. Adder
  - d. Latches
- B. Arquitectura Memorias
  - 1. Representación de valores multi-bit
  - 2. Registros
  - 3. Direccionamiento
- C. Máquinas de Estado
  - 1. Combinadas versus secuenciales
  - 2. Diagrama de estados
  - 3. Máquinas de estados finitos
- D. Circuitos Integrados y Flip-Flops
- IV. Aplicación de Circuitos
  - A. Circuitos en acción
  - B. Von Neuman Model
    - 1. Memoria
    - 2. Unidad de procesamiento
    - 3. Entrada y salida
    - 4. Unidad de control
    - 5. Procesamiento de las instrucciones
    - 6. Fetch, Decode, Evaluate, Operands, Execute, Store
    - 7. Diagrama LC-3
- V. Introducción Programación Assembler
  - A. Programación Assembler
    - 1. ¿Qué es assembler?
    - 2. Sintaxis lenguaje LC-3
    - 3. Op Codes y operandos
    - 4. Etiquetas y comentarios
    - 5. Directivas

6. "Trap codes"
  - B. Proceso de compilación y ensamblaje
    1. Propósito
    2. Primera pasada
    3. Segunda pasada
    4. Código objeto (.obj)
    5. Programar para el emulador LC-3
    6. "Linking and Loading"
  - C. "Tracing" y "Debuging"
    1. Operaciones
    2. Niveles
    3. Herramientas
    4. Tipos de errores
    5. "Tracing"
  - D. Creación programa básico para LC-3
- VI. Programación Assembler con LC-3
- A. Entrada y salida de datos (I/O)
    1. Tasas de transferencias equipos I/O
    2. El controlador de I/O
    3. Registros y "DMA" (Acceso Memoria Directa)
    4. Rutina simple de lectura de una letra del teclado
    5. Salida a la pantalla
  - B. Interrupciones y "polling"
    1. Métodos de entrada de datos
    2. Transferencia de control
      - a. "Polling"
      - b. Interrupciones
      - c. Prioridades
  - C. Creación programa con entrada de datos
  - D. Creación de programa con entrada de datos, cálculos, almacenaje y lectura de datos de memoria, procesamiento de datos y salida de datos a pantalla

## **METODOLOGÍA**

Se recomiendan las siguientes estrategias de la metodología de aprendizaje activo:

Conferencias  
Informes escritos  
Proyectos de investigación/creación/aplicación/acción  
Simulaciones  
Trabajos colaborativos o grupales  
Informes escritos  
Ejercicios de búsqueda de información  
Laboratorio obligatorio independiente  
Uso de Web para comunicaciones y conferencias  
Uso de recursos disponibles en la Internet

## **RECURSOS**

Computadora con Acceso a la Internet.  
Programa Emulador Logicly.  
Programa Emulador LC3.  
Laboratorio de Computadoras con acceso a configuraciones de los equipos.

## **EVALUACIÓN**

Exámenes parciales (pruebas cortas y final)	35%
Tareas y micro-investigaciones (personajes)	15%
Laboratorios y programas parciales	25%
Proyecto de programación (final)	<u>25%</u>
<b>Total</b>	<b>100%</b>

## **AVALÚO DEL APRENDIZAJE**

Se aplica la rúbrica de avalúo institucional a la actividad central del curso.

### **RÚBRICA PROYECTOS PROGRAMACIÓN**

50 puntos por que entregue/funcione  
15 puntos por eficiencia y originalidad del programa  
15 puntos por calidad de interacción con el usuario  
10 puntos por calidad documentación adicional  
5 puntos por calidad documentación dentro del programa  
5 puntos por corrección ortográfica y gramatical

## **BIBLIOGRAFÍA**

### **TEXTO**

Patt, Y., Sanjay, J. P. (2004) *Introduction to computing systems*. Boston: McGraw-Hill Higher Education.

## REFERENCIAS

- Bryant, R. E, & O'Hallaron, D. (2011). *Computer systems*. Boston: Prentice Hall.
- Duntemann, J. (2009). *Assembly language step-by-step*. Indianapolis, IN: Wiley Pub.
- Monk, S. (2014). *Raspberry Pi cookbook*. Sebastopol, CA: O'Reilly Media.
- Rainer, R. K. & Cegielski, C. G. (2011). *Introduction to information systems*. Hoboken, N.J.: Wiley.
- Smith, B. (2014). *Raspberry Pi assembly language*. Sydney, Australia: BSB.

## BIBLIOTECA VIRTUAL

Para utilizar la Biblioteca Virtual siga los siguientes pasos:

### Para acceder desde Sagrado:

- escriba la dirección <http://biblioteca.sagrado.edu/>
- seleccione Biblioteca Virtual y aparecerá la página en la cual podrá comenzar a hacer sus búsquedas de información y acceder a las bases de datos. Puede acceder a las bases de datos por disciplina o por orden alfabético.

### Para acceder desde fuera de la Universidad o utilizando computadoras personales:

- escriba la dirección <http://mi.sagrado.edu>
- escriba su nombre del usuario y la contraseña
- seleccione el ícono de **Biblioteca Virtual**
- aparecerá la página en donde podrá comenzar a hacer sus búsquedas de información y acceder a las bases de datos. Puede acceder a las bases de datos por disciplina o por orden alfabético.

En la Biblioteca Virtual encuentran el contenido de artículos publicados en revistas profesionales especializadas en el tema, ejemplo de algunas son: Communications of the ACM, IT Professional, Journal of Computing & Information Technology, Journal of Communications Software & Systems.

## DIRECCIONES ELECTRÓNICAS

<http://logic.ly/>

<http://logic.ly/lessons/>

<http://paulbourke.net/dataformats/bitmaps/>

<http://www.csfieldguide.org.nz/DataRepresentation.html>

<http://voyager.egglescliffe.org.uk/mwc/mukoku/mod/resource/view.php?id=518>

[http://highered.mheducation.com/sites/0072467509/student\\_view0/lc-3\\_simulator.html](http://highered.mheducation.com/sites/0072467509/student_view0/lc-3_simulator.html)

<http://cs.brynmawr.edu/Courses/cs240/fall2013/highlevelprogramming.html>

<https://www.cs.auckland.ac.nz/courses/compsci210s2c/tutorials/LC3WinGuide.pdf>  
[http://en.wikipedia.org/wiki/Carry-lookahead\\_adder](http://en.wikipedia.org/wiki/Carry-lookahead_adder)  
[http://en.wikipedia.org/wiki/Adder\\_\(electronics\)#Ripple\\_carry\\_adder](http://en.wikipedia.org/wiki/Adder_(electronics)#Ripple_carry_adder)  
[http://en.wikipedia.org/wiki/Dennis\\_Ritchie](http://en.wikipedia.org/wiki/Dennis_Ritchie)  
[http://en.wikipedia.org/wiki/Ken\\_Thompson](http://en.wikipedia.org/wiki/Ken_Thompson)  
[http://en.wikipedia.org/wiki/Gordon\\_Moore](http://en.wikipedia.org/wiki/Gordon_Moore)  
[http://en.wikipedia.org/wiki/Moore's\\_law](http://en.wikipedia.org/wiki/Moore's_law)

Puede encontrar más recursos de información relacionados a los temas del curso en la página de la biblioteca <http://biblioteca.sagrado.edu/>

## **ACOMODO RAZONABLE**

Para obtener información detallada del proceso y la documentación requerida, debe visitar la oficina correspondiente. Para garantizar igualdad de condiciones, en cumplimiento de la Ley ADA (1990) y el Acta de Rehabilitación (1973), según enmendada, todo estudiante que necesite servicios de acomodo razonable o asistencia especial deberá completar el proceso establecido por la Vicepresidencia de Asuntos Académicos.

## **INTEGRIDAD ACADÉMICA**

Esta política aplica a todo estudiante matriculado en la Universidad del Sagrado Corazón para tomar cursos con o sin crédito académico. Una falta de integridad académica es todo acto u omisión que no demuestre la honestidad, transparencia y responsabilidad que debe caracterizar toda actividad académica. Todo estudiante que falte a la política de honradez, fraude y plagio se expone a las siguientes sanciones: recibirá nota de cero en la evaluación y/o repetición del trabajo en el seminario, nota de F(\*) en el seminario: suspensión o expulsión según se establece en el documento de Política de Integridad Académica con fecha de efectividad de noviembre 2022.

## **CURSOS DE INVESTIGACIÓN**

“Este curso puede requerir que los estudiantes practiquen tareas relacionadas al proceso de investigación, tales como: toma de consentimiento o asentimiento informado, administración de instrumentos, realización de entrevistas, observaciones o grupos focales, entre otros. Estas tareas son parte de un ejercicio académico y no se utilizara la información recopilada para compartirla con terceros o divulgarla en otros escenarios que no sean el salón de clases junto al profesor que enseña el curso. Todo estudiante que vaya a interactuar con sujetos humanos como parte de su práctica en investigación tiene que estar certificado en ética con sujetos humanos en la investigación por el *Collaborative Institutional Training Initiative (CITI Program)*, al igual que su profesor”(Los cursos de investigación deben incluir este texto)



Derechos reservados | Sagrado | Noviembre, 2022 (2015)