

| PROGRAMA | | | |
|------------------------------------|---|-----------------|-------------|
| Espacio Curricular: | Programación en C | | |
| Carácter: | Electivo | Período: | 2º Semestre |
| Carrera/s: | Licenciatura en Ciencias Básicas (Orientación Biología, Geología, Física, Matemática y Química) Profesorado en Ciencias Básicas (Orientación Física, Matemática y Química) | | |
| Profesor Responsable: | Emmanuel N. Millán | | |
| Equipo Docente: | | | |
| Carga Horaria: | 80 hrs; 6 horas semanales: 2 de teoría y 4 de práctica. | | |
| Recomendaciones de Cursado: | Informática | | |

1-EXPECTATIVAS DE LOGRO

- Dominar los conceptos básicos del lenguaje C
- Identificar y organizar los datos pertinentes al problema, evaluando el contexto particular.
- Comprender las ventajas y desventajas del lenguaje de C.
- Conocer las fuentes de información disponibles para que el/la estudiante pueda seguir mejorando sus habilidades.
- Aplicar el lenguaje para resolver problemas relacionados a la especialidad del estudiante.
- Ser capaz de relacionar el conocimiento profundo de las ciencias básicas con la resolución de problemas a través del lenguaje de programación C.
- Aplicar el pensamiento lógico, científico y sistémico sobre problemas de ciencias básicas.

2-DESCRIPTORES

El uso de C para el procesamiento y análisis de datos. Estructuras de datos, controles condicionales y de bucles, funciones, punteros, arreglos, entrada y salida de archivos. Programación estructurada y modular, software de compilación, depuración y manipulación de código.

3-APORTE DEL ESPACIO CURRICULAR A LA CONSTRUCCIÓN DEL PERFIL, ALCANCE Y ACTIVIDADES RESERVADAS DEL TÍTULO

Este espacio curricular está íntimamente vinculado al perfil de la Licenciatura en Ciencias Básicas, dado que apunta a que los egresados sean idóneos en el manejo de datos y herramientas informáticas. Dada la importancia que tienen hoy en día la informática y la computación en nuestras vidas cotidianas, es indispensable brindar a los alumnos de la Licenciatura en Ciencias Básicas una debida formación en informática, para que al egresar puedan responder a las demandas reales del sector científico y/o socio-productivo.

En particular, el lenguaje C es una valiosa herramienta de cálculo y de análisis. Este lenguaje es de amplio uso en el área académica y científica debido a su alto rendimiento. Todas las unidades de este programa están acompañadas de prácticas con ejercicios relacionados a las ciencias básicas.

Este espacio curricular contribuye a:

- 1) idear o perfeccionar las aplicaciones industriales, médicas y de otra índole de los principios de la disciplina indicada en la orientación,
- 2) efectuar, evaluar experimentos, ensayos y análisis para investigar los efectos de actividades vinculadas a la disciplina sobre el medio ambiente, asesorando a especialistas en este campo e idear o perfeccionar materiales y productos.
- 3) trabajar en proyectos interdisciplinarios que involucren las Ciencias Exactas y Naturales
- 4) generar conocimiento científico tanto para su aplicación en problemas concretos, de carácter básico o aplicado, como para la transmisión del saber en las Ciencias Exactas y Naturales

4-COMPETENCIAS

| Tipo de Competencias | Detalle | Articula con: |
|----------------------|---|---|
| Genéricas | <ol style="list-style-type: none"> 1. Manejo de herramientas informáticas para el análisis y visualización de datos. 2. Resolución numérica de problemas concretos en biología, física y química mediante herramientas informáticas | <ol style="list-style-type: none"> 1) Taller de física computacional, Bioinformática, Aplicaciones de Ecuaciones Diferenciales. 2) Introducción de las simulaciones numéricas en las ciencias básicas. 3) Programación en Python |
| Específicas | <ol style="list-style-type: none"> 1. Comprensión lógica de un programa de computación. | <ol style="list-style-type: none"> 1) Programación en Python |

5-CONTENIDOS ANALÍTICOS

Unidad 1: Fundamentos de lenguajes de programación

Lenguajes de programación, traductores y compiladores. La compilación y sus fases. Paradigmas de programación (imperativos, declarativos, orientados a objetos). Metodología de programación (modular, orientada a objetos, estructurada). Diseño de algoritmos, escritura de algoritmos. Diagramas de flujo. Pseudocódigo. Herramientas de programación: IDE, compilador, depurador, make/cmake.

Unidad 2: Introducción a C

Estructura general. Creación de un programa básico. Depuración (errores de sintaxis, lógicos, regresión). Mensajes de error, errores en tiempo de ejecución. Elementos de un programa de C. Tipos de datos C. Variables, alcance. Entrada y salida básica. Operadores. Preprocesador. Formatos de entrada y salida de flotantes, enteros, cadenas.

Unidad 3: Estructuras de selección y control:

Condicionales if y switch. Bucles while, for, do while. Bucles infinitos. Break y continue

Unidad 4: Funciones:

Estructura de una función. Prototipos de las funciones. Parámetros por valor y referencia. Ámbitos: de una variable, del programa, archivo fuente, función y bloque. Funciones de bibliotecas. Funciones numéricas, matemáticas, trigonométricas, generación de números aleatorios. Funciones recursivas.

Unidad 5: Arreglos, caracteres y cadenas:

Declaración, ejemplos, índices. Arreglos multidimensionales. Pasaje de arreglos a funciones. Concepto de cadena de caracteres. Lectura de cadenas. Biblioteca string.h. Funciones varias para el tratamiento de cadenas

Unidad 6: Punteros:

Direcciones de memoria. Concepto de puntero. Operadores para punteros. Llamada a funciones por referencia. Relación entre punteros y arreglos. Arreglos de apuntadores. Asignación de memoria dinámica

Unidad 7: Estructuras y uniones:

Definición y uso de estructuras. Estructuras anidadas. Arreglos a estructuras. Uniones

Unidad 8: Procesamiento de Archivos:

Flujos, apuntador a archivo. Apertura y cierre de un archivo. Funciones de entrada/salida para archivos. Archivos binarios. Funciones para acceso aleatorio. Datos externos al programa (argumentos de main)

Unidad 9: Introducción a algoritmos y estructuras de datos avanzadas:

Listas enlazadas, pilas, colas y árboles. Algoritmos de ordenación y búsqueda. Conceptos básicos de arquitectura de computadoras. Ejemplo de paralelización básica con OpenMP. Pruebas de rendimiento.

6-BIBLIOGRAFÍA

1. Joyanes Aguilar, L., & Zahonero Martínez, I. (2014). Programación en C, C++, JAVA y U ML. México DF: Mc. Graw-Hill. Segunda Edición
2. Mathieu, M. J. (2014). Introducción a la programación. Grupo Editorial Patria.
3. Kernighan, B. W., & Ritchie, D. M. (1991). *El lenguaje de programación C*. Pearson Educación.
4. Deitel, H. M., & Deitel, P. J. (2004). *Cómo programar en C/C++ y Java*. Pearson educación.
5. Perry, G. M., & Miller, D. (2014). *C programming: absolute beginner's guide*. Pearson Education.
6. Shaw, Z. A. (2015). *Learn C the hard way: practical exercises on the computational subjects you keep avoiding (like C)*. Addison-Wesley Professional.
7. Seacord, R. C. (2020). *Effective C: An Introduction to Professional C Programming*. No Starch Press.
8. Prata, S. (2014). *C primer plus*. Pearson Education.
9. Griffiths, D., & Griffiths, D. (2012). *Head First C: A Brain-Friendly Guide*. O'Reilly Media, Inc..

7-METODOLOGÍA DE ENSEÑANZA

El contenido de la materia será introducido mediante presentaciones orales a cargo del docente. Se podrán utilizar herramientas de comunicación virtual para realizar clases de teoría y/o práctica de forma no-presencial. Todos los materiales estarán disponibles en el aula virtual. Además, en la medida que la situación sanitaria lo permita, se realizarán clases prácticas en la sala de computación, donde los estudiantes aplicarán los conocimientos adquiridos resolviendo guías de problemas. En caso de ser necesario por la situación sanitaria, el espacio curricular está preparado para ser dictado 100% virtual

8- EVALUACIÓN DURANTE EL CURSADO Y CONDICIONES DE REGULARIDAD

Los asistentes serán evaluados de manera continua durante el desarrollo de los aspectos prácticos de las clases, y mediante una evaluación integradora al finalizar el cursado.

Para poder acceder a dicha evaluación integradora, el asistente deberá tener un mínimo de 70% de asistencia al curso. La evaluación integradora tomará la forma de ejercicios computacionales, en los cuales los asistentes deberán crear o completar programas informáticos en C para resolver problemas específicos. El acceso al material del curso e internet no se restringirá durante la evaluación, ya que refleja un entorno de trabajo realista. Los ejercicios enviados se evaluarán en función de su capacidad para resolver el problema, la eficiencia del código, la legibilidad del código y la calidad de salida (cuando corresponda)

Para obtener la regularidad del espacio curricular la nota de esa evaluación deberá ser mayor o igual a 6 (seis).

9- SISTEMA DE APROBACIÓN FINAL Y/O PROMOCIÓN DEL ESPACIO CURRICULAR

Para promocionar el espacio curricular se debe tener una nota global (nota de la evaluación integradora y prácticas) mayor o igual a 8 (ocho) y cumplir con la asistencia del 70%.

Para regularizar el espacio curricular se debe tener una nota global (nota de la evaluación integradora y prácticas) mayor o igual a 6 (seis) y cumplir con la asistencia del 70%.

Modalidad de evaluación para estudiantes libres

- ❖ Aprobar, con una nota de 7 (siete) o mayor, una evaluación práctica que abarca todo el material del curso.
- ❖ Aprobar, con una nota de 7 (siete) o mayor, una evaluación oral para determinar el nivel de comprensión del material del curso.
- ❖ Presentar y aprobar todos los trabajos prácticos antes de rendir la evaluación oral y escrita.

PROMOCIONABLE (Marque con una cruz la respuesta correcta)

SÍ

X

NO

10- LINEAMIENTOS DE INVESTIGACIÓN DE LOS INTEGRANTES DEL ESPACIO CURRICULAR

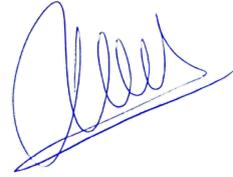
11- INFRAESTRUCTURA Y EQUIPAMIENTO

1. Para que el curso sea efectivo, el profesor requerirá el uso de la sala de informática seis horas por semana, divididas en dos clases de tres horas.
2. Debido a los recursos limitados en el punto uno (1) habrá un cupo máximo de asistentes. Se dará prioridad a los solicitantes de FCEN-UNCuyo, seguidos por los estudiantes vocacionales de otras facultades de UNCuyo y, finalmente, de los solicitantes que no estén asociados con UNCuyo.
3. Para enseñar el curso de manera efectiva, el profesor requerirá el uso de un proyector digital, y que una distribución de Linux tenga instalado el compilador GCC y un editor de textos en cada una de las máquinas en la sala de computación.


Emmanuel Millan

***FIRMA Y ACLARACIÓN
DEL RESPONSABLE DEL ESPACIO CURRICULAR***

✓



Alejandro M. Lobos
Director Orientación Física