

Aspectos generales

Título:	Principios de Programación
Programas de posgrado o planes de estudio en donde se ofertará adicionalmente:	
Posgrado en Ciencias Biomédicas Posgrado en Ciencias Bioquímicas Posgrado en Ciencias Biológicas	
Área del conocimiento:	Genética, genómica y bioinformática
Semestre:	2027-1
Modalidad:	Curso fundamental
Horario:	Martes y jueves 12:00 a 14:00 hrs
No. sesiones:	32
Horas por sesión:	2.0
Total alumnos PDCB:	6
Total alumnos:	6
Videoconferencia:	Si
Lugar donde se imparte:	Aulas de la Licenciatura en Ciencias Genómicas, UNAM campus Morelos
Informes:	jfreyre@ccg.unam.mx

Métodos de evaluación

MÉTODO	PORCENTAJE	NOTAS
Actividades	25%	Tareas asignadas para realizar en clase/casa/asesoría
Exámenes	30%	Exámenes de evaluación de conocimientos
Participación	15%	Participación oral del estudiante en clase/asesoría
Proyecto final	30%	Proyecto final donde el estudiante ejercitará y demostrará los conocimientos aprendidos

Contribución de este curso/tópico en la formación del alumnado del PDCB:

La explosión masiva de datos biológicos ha ocasionado que los análisis de frontera en genómica posean un fuerte componente de las ciencias de la información. Esto hace fundamental que un estudiante de doctorado se prepare y desarrolle habilidades dentro de las ciencias computacionales, las cuales le permitan procesar la información que requerirá y producirá a lo largo de su investigación y vida profesional.

El objetivo principal de este curso es desarrollar en el alumno la lógica necesaria para aprender a diseñar algoritmos y programas computacionales de forma independiente del lenguaje. Esta independencia del lenguaje lo preparará para que en su vida profesional pueda desarrollar programas en cualquier lenguaje de programación requerido dentro del paradigma procedural o imperativo que es el más ampliamente usado.

Profesor (a) responsable

Nombre:	Freyre González Julio Augusto
Teléfono:	(777) 329-1777 Ext. 38483
Email:	jfreyre@ccg.unam.mx

Profesores (as) participantes

PARTICIPANTE	ENTIDAD O ADSCRIPCIÓN	SESIONES
--------------	-----------------------	----------

FREYRE GONZÁLEZ JULIO AUGUSTO Centro de Ciencias Genómicas
Responsable

Estructuras de datos
Estructuras de datos
Estructuras de datos
Estructuras de datos
Estructuras de datos
Estructuras de datos
Estructuras de datos
Estructuras de datos
Estructuras de datos
Estructuras de datos
Introducción a la computación
Introducción a la computación
Introducción a la computación
Introducción a la computación
Introducción a la computación
Introducción a la computación
Módulos (subrutinas, funciones, procedimientos)
Módulos (subrutinas, funciones, procedimientos)
Módulos (subrutinas, funciones, procedimientos)
Módulos (subrutinas, funciones, procedimientos)
Programación estructurada
Programación estructurada
Programación estructurada
Programación estructurada
Programación estructurada
Programación estructurada
Programación estructurada
Programación estructurada
Programación estructurada
Programación estructurada
Programación estructurada
Programación estructurada
Programación estructurada

Introducción

El objetivo principal de este curso es desarrollar en el alumno la lógica necesaria para aprender a diseñar algoritmos y programas computacionales de forma independiente del lenguaje. Esta independencia del lenguaje lo preparará para que en su vida profesional pueda desarrollar programas en cualquier lenguaje de programación requerido dentro del paradigma procedural o imperativo que es el más ampliamente usado.

Para lograr esto, emplearemos un enfoque distinto al que se emplea en los cursos tradicionales de programación. En consecuencia, en este curso no ortodoxo, el alumno aprenderá los fundamentos de la computación y la programación, y deberá desarrollar una forma de razonamiento abstracto que le permita conceptualizar los problemas y sus soluciones alternas.

En muchos aspectos, la forma de resolver computacionalmente problemas en genómica, y en la ciencia en general, no es muy diferente a cómo se hace en otras áreas no científicas. Sin embargo, existe un componente distintivo muy importante: en investigación generalmente no sabemos la respuesta correcta a priori, por ello contar con la habilidad de desarrollar código computacional confiable, claro, no ambiguo y auto-consistente es fundamental en ciencia. Así, el alumno aprenderá a, dado un problema particular, identificar cual es el lenguaje más óptimo de acuerdo a sus características, desarrollar la solución a su problema empleando la metodología adecuada, validar su algoritmo y detectar errores, así como se enfrentará a nuevos retos al implementar en una computadora sus soluciones.

Temario

Introducción a la computación (6 sesiones, 12 horas)
¿Qué es una computadora?
¿Qué es un algoritmo?
¿Qué es un lenguaje de programación?
¿Qué tipos de lenguajes de programación existen?
¿Qué es un paradigma de programación y cuáles existen?
¿Cuál es la metodología para solucionar problemas con una computadora?
Programación estructurada (12 sesiones, 24 horas).
Variables
Estructuras secuenciales
Operadores
Estructuras de control de flujo
Módulos (subrutinas, funciones, procedimientos) (4 sesiones, 8 horas).
Ámbito de una variable (variables globales y locales)
Diferencia entre funciones y procedimientos
Paso de argumentos por valor y por referencia
Recursividad
Estructuras de datos (10 sesiones, 20 horas).

Arreglos lineales (vectores: `<tipo> <id>[<dim>]`)
Arreglos no lineales

Bibliografía

Wirth, N. Algoritmos y Estructuras de Datos. Prentice Hall, 1987.
Lipschutz, S. Estructura de Datos. Serie Schaum en Computación. McGraw-Hill, 1988.
Wirth, N. Algoritmos + Estructuras de Datos = Programas. Dossat, 1992.
Dijkstra, E.W. A Discipline of Programming. Prentice Hall, 1997.
Knuth, D.E. The Art of Computer Programming Vol. 1, 2, and 3 (3rd Ed.). Addison Wesley, 1997.
Aho, A.V., Hopcroft, J.E., and Ullman, J.D. Data Structures and Algorithms. Addison-Wesley, 1983.
Sudkamp, T. A. Languages and Machines: An Introduction to the Theory of Computer Science (2nd Ed.). Addison-Wesley, 1998.
Hopcroft, J. E., Motwani, R. and Ullman, J. D. Introduction to Automata Theory, Languages and Computation (2nd Ed.). Addison-Wesley Longman, 2000.
Kernighan, B.W., y Ritchie, D.M. El Lenguaje de Programación C (2da Ed.). Prentice Hall, 1991.
Schwartz, R.L., and Phoenix, T. Learning Perl (3rd Ed.). O'Reilly, 2001.
Wall, L., Christiansen, T., and Orwant, J. Programming Perl (3rd Ed.). O'Reilly, 2000.
Biermann, A.W. Great Ideas in Computer Science. The MIT Press, 1997.
Ralston, A., Reilly, E. D. and Hemmendinger, D. Encyclopedia of Computer Science (4th Ed.). Nature Publishing Group, 2000.

Observaciones

Al ser este un curso fundamental la bibliografía comprende principalmente textos clásicos que siguen siendo básicos para la enseñanza de esta materia.

SUGERENCIA: El método actual para asignar los participantes a los temas y sesiones es bastante tedioso, sería mejor si permitiera introducir rangos de sesiones (eg, 5-10) en lugar de meter cada integrante en cada una de las sesiones.