
Nombre de la institución: Instituto De Enseñanza Superior N° 6017 «Prof. Amadeo R. Sirolli».

Carrera: Profesorado de Educación Secundaria en Informática

Plan de estudios: Aprobado por Resolución Ministerial N° 1168/16

Año lectivo: 2023

Campo: Campo de la Formación Específica

Curso: 2° año **División:** 1ra

Asignatura: Lenguaje de Programación I: Programación Imperativa

Docente Responsable: Rodolfo Ezequiel Baspineiro

Régimen de la asignatura: Anual

Cantidad de horas-cátedra: 5 horas

Condiciones para regularizar la asignatura

- **70% de asistencia obligatoria a clases teóricas y prácticas y a las actividades complementarias previstas por el docente responsable de la cátedra.** Este porcentaje podrá reducirse al 60% cuando las ausencias obedezcan a razones de salud, de trabajo o de fuerza mayor debidamente justificadas por el Consejo Asesor. Art. 36° - RAM 2484/2013. Al reducirse el porcentaje de asistencias al 60% por las razones mencionadas, el profesor podrá realizar una evaluación integradora que incluya los temas abordados durante el período lectivo correspondiente para regularizar la asignatura en cuestión. En caso de no aprobarse el mencionado examen, el estudiante quedará en condición de libre. RAM 2484/2013 – Art. 37°.
- **100% de aprobación de trabajos prácticos. 5 trabajos prácticos.**
- **100% de parciales aprobados. 2 (dos) con nota mínima 6 (seis). Con sus instancias recuperatorias correspondientes.**

Condiciones para rendir como alumno regular

- Cumplimentar las condiciones de regularidad al momento del examen.
- Inscribirse en tiempo y forma para el examen.

-
- El estudiante deberá presentar indefectiblemente ante el tribunal evaluador: Libreta de estudiante, Documento Nacional de Identidad o documento equivalente y fotocopia del programa de la unidad curricular para cual se presenta. Art. 49° - RAM 2484/2013.
 - Examen oral según Art. 43° - RAM 2484/2013.

Para aprobar el examen final los estudiantes deberán obtener como mínimo una calificación de 4 (cuatro) puntos en una escala de 1 (uno) a 10 (diez) puntos, la que será expresada en números enteros. Art. 55° - RAM 2484/2013.

Condiciones para rendir como alumno libre

- Inscribirse en tiempo y forma para el examen.
- El estudiante deberá presentar indefectiblemente ante el tribunal evaluador: Libreta de estudiante, Documento Nacional de Identidad o documento equivalente y fotocopia del programa de la unidad curricular para cual se presenta. Art. 49° - RAM 2484/2013.
- Debe rendir una evaluación oral y una escrita, debiéndose archivar el escrito en la institución como constancia según Art. 43° - RAM 2484/2013.

Para aprobar el examen final los estudiantes deberán obtener como mínimo una calificación de 4 (cuatro) puntos en una escala de 1 (uno) a 10 (diez) puntos, la que será expresada en números enteros. Art. 55° - RAM 2484/2013.

Marco Teórico

El espacio curricular Lenguaje de Programación I: Programación Imperativa es del segundo año de la carrera siendo el primer espacio curricular en donde el estudiantado procede a realizar sus primeras prácticas de programación mediante un lenguaje de programación de alto nivel del paradigma imperativo/estructurado.

La asignatura al ser orientada a la programación estructurada, requiere de una fundamentación teórica respecto al paradigma imperativo, en donde se crean procedimientos y se organizan sentencias de programación mediante buenas prácticas que brinda la ingeniería en sistemas, esta cátedra hace suya la postura del autor Niklaus Wirth

plasmada en una de sus mayores obras literarias: Algoritmos + Estructuras de datos = Programas.

Estos conocimientos serán la base del pensamiento computacional desarrollado por cada estudiante, permitirá escalar al paradigma orientado a objeto en la materia Lenguaje de Programación II Programación Orientada a Objetos.

Es un espacio curricular con combinación contenidos teóricos y prácticos, el mismo tiene un enfoque teórico/práctico a lo largo de toda la cursada, favoreciendo las instancias en las cuales los estudiantes puedan construir y demostrar su comprensión de los contenidos.

Propósitos

- Aprender conceptos fundamentales de un lenguaje de programación de alto nivel.
- Aplicar el uso de estructuras de datos, de control y variables
- Articular diversas estructuras para lograr implementar un algoritmo.
- Diseñar propuestas educativas con software orientado a la enseñanza en el nivel medio.

Objetivos generales

- Utilizar la terminología específica de la asignatura.
- Demostrar dominio teórico y práctico de la programación imperativa
- Diagramar soluciones computacionales en software Scratch
- Reconocer estructuras articuladas entre sí de un lenguaje de programación imperativo
- Resolver situaciones problemáticas computacionales mediante la creación de programas con salidas de textos.

Contenidos

Unidad I

Problemas básicos con diagramas de bloques. Algoritmos, Pseudocódigo. Estructuras de control y datos básicas. Listas de tamaño definidos. Programación en papel. Software de aprendizaje de programación interactivos (Lightbot, Pilas bloques y Scratch).

Unidad II

Diagramas de flujo. Programación estructurada. Encabezamiento, bloque y cuerpo. Declaraciones y definiciones. Tipos de datos estándar: enteros, reales, caracteres y lógicos. Sentencia de asignación, de entrada, de salida y salidas formateadas. Operaciones mediante tipos de datos.

Unidad III

Estructuras de control. Bifurcación o selección simple y selección múltiple (alternativas). Repetición o iteración: con cantidad conocida de veces y con cantidad desconocida de veces (ciclo condicional e incondicional). Vectores y matrices. Arreglos. Ordenamiento de un arreglo unidimensional. Búsqueda de un valor en un arreglo unidimensional: búsqueda secuencial y búsqueda dicotómica

Unidad IV

Tipos de datos no estándar o definidos por el programador. Tipo enumerado o escalar, tipo subrango o intervalo. Tipo estructura: arreglos unidimensionales y multidimensionales. Ordenamiento y búsqueda en un arreglo bidimensional.

Unidad V

Recursividad: definición, traza. Algoritmos recursivos. Métodos de ordenación y búsqueda recursivos. Comparaciones con versiones iterativas. Tipos de datos dinámicos. Manejo de la memoria para asignación dinámica, utilidades de tipos de datos dinámicos, asignación y liberación en tiempo de ejecución. Arreglos con tipos de datos dinámicos. Resolución de situaciones problemáticas.

Metodología de trabajo

La asignatura corresponde al segundo año de la carrera del profesorado con un régimen de tipo anual, siendo una asignatura con fuerte contenido técnico incorporando metodologías de desarrollo con lenguajes de programación, la propuesta áulica incluye

clases teóricas/prácticas en las que se realizará una recopilación abreviada de los contenidos vistos durante la clase anterior a los fines de continuar con la secuencialidad, luego se presentarán los nuevos contenidos teóricos continuados con su aplicación práctica, con instancias con fuerte presencia de dispositivos informáticos al ser necesario para realizar el desarrollo de aplicaciones, se trabajarán problemáticas vistas en años anteriores y en aquellas con un vínculo de la enseñanza de la temática en el nivel medio.

Al estar en segundo año, se espera contribuir en la formación de la comunicación argumentada mediante conceptos teóricos en debates entre grupos y a nivel comisión, para esto los trabajos prácticos continuarán con el enfoque teórico/práctico con actividades basadas en situaciones problemáticas y aplicación de los contenidos, logrando que los estudiantes puedan tomar decisiones con la menor incertidumbre y la correcta fundamentación sin dejar de lado el futuro rol de docentes que desempeñarán los y las estudiantes.

Cada unidad de la materia dispondrá de una serie de actividades reflexivas de un posible ambiente áulico en donde se esté enseñanza lenguajes de programación del paradigma imperativo que despierten curiosidad, de esta forma podemos recurrir a analogías para entablar debates al momento de pensar las soluciones y posteriormente realizar algoritmos juntos con sus diagramas.

Se utilizará la plataforma Educativa para brindar un espacio de intercambio mediante foros virtuales y la posibilidad de revisar conceptos de programación con el entorno de programación. Por último, los estudiantes podrán utilizar software ofimático en línea los cuales se encuentra disponibles gratuitamente mediante las cuentas de correo Gmail o Outlook dependiendo si es Google Docs o Office 365 respectivamente.

Evaluación

La evaluación constituye el componente necesario para analizar el proceso de enseñanza y aprendizaje, para ello se recurre a la utilización de tres tipos de evaluación: diagnóstica, formativa o proceso y sumativa. Cada una constituye un insumo importante para emitir juicios de valor.

Se realizarán rubricas correspondientes para monitorizar y acompañar las trayectorias académicas, con especial énfasis en el cumplimiento de plazos, a los fines de alcanzar los requisitos de aprobación y la creación de hábitos de trabajos que les serán útiles para futuras materias de la carrera.

Cuando los estudiantes no hubieran alcanzado en los exámenes parciales la calificación de APROBADO o registraran ausente por razones debidamente justificadas, tendrán derecho a un examen recuperatorio por parcial. Art. 41º - RAM 2484/2013

Tanto el primer parcial como el segundo parcial será llevado a cabo en articulación con la materia Didáctica de la Informática para la Escuela Secundaria en donde se compartirá una misma exposición en la cual ambos docentes evaluarán los contenidos correspondientes a cada espacio curricular, en la medida de lo posible se utilizarán los recursos de la institución incluyendo netbooks y proyector para la realización de las exposiciones.

Criterios de evaluación

- Uso correcto de terminología de la asignatura
- Articulación de las estructuras de datos y de control para resolver problemas mediante un software de programación.
- Resolución de diagramas mediante software de programación
- Creación de algoritmos que puedan ser aplicados mediante software de programación
- Manipulación de vectores unidimensionales y bidimensionales
- Toma de decisiones justificadas.
- Exposición frente a pizarra/pizarrón
- Demostración de trabajo colaborativo.
- Análisis de diferentes situaciones problemáticas en tiempos apropiados.

Criterios para mesas examinadoras

Se utilizan los mismos criterios de evaluación definidos para el cursado con la particularidad de concentrarse en la exposición oral justificada con contenido teórico. A continuación, las modalidades para las mesas examinadoras

Para estudiantes en calidad de REGULAR

- Examen oral de 15 minutos exponiendo una situación problemática resuelta con un lenguaje de programación imperativo indicando una breve presentación del

problema, su solución y conclusión sobre el motivo de su abordaje en una actividad del nivel medio.

- Posterior al examen oral, unos minutos para preguntas aclaratorias por parte del tribunal.

Para estudiantes en calidad de LIBRE

- Examen escrito de 40 minutos en donde la o el estudiante deberá resolver dos ejercicios, el primero mediante diagrama y caso de prueba mientras que el segundo mediante código, en caso de no aprobar la instancia escrita, no pasará a la instancia oral quedando la o el estudiante desaprobado.
- Examen oral de 15 minutos exponiendo una situación problemática resuelta con un lenguaje de programación imperativo indicando una breve presentación del problema, su solución y conclusión sobre el motivo de su abordaje en una actividad del nivel medio.
- Posterior al examen, unos minutos para preguntas aclaratorias por parte del tribunal.

Bibliografía Específica por Unidad

<p>Unidad N° 1</p>	<ul style="list-style-type: none"> • López García J. C. (2011) Programación con Scratch. Fundación Gabriel Piedrahita Uribe. Recuperado de https://eduteka.icesi.edu.co/pdfdir/AlgoritmosProgramacionCuaderno1.pdf • Factorovich, P. (2015) Actividades para aprender a Program.AR: segundo ciclo de educación primaria y primero de la secundaria. Fundación Sadosky. Recuperado de https://www.fundacionsadosky.org.ar/wp-content/uploads/2015/08/Actividades%20para%20aprender%20a%20Programar.pdf • Presentaciones digitales desarrolladas por el docente. • Documentación desarrollada por el docente.
<p>Unidad N° 2</p>	<ul style="list-style-type: none"> • Kernighan B, Ritchie D. (1991). El lenguaje de programación C. Pearson Education. Recuperado de https://frrq.cvg.utn.edu.ar/pluginfile.php/13741/mod_resource/content/0/El-lenguaje-de-programacion-C-2-ed-kernighan-amp-ritchie.pdf Capítulo 1 y 2 • Presentaciones digitales desarrolladas por el docente. • Documentación desarrollada por el docente.

	<ul style="list-style-type: none"> Banahan M, Brady D, Doran M (2019) The C book. Addison Wesley. Recuperado de https://publications.gbdirect.co.uk//c_book/ (Solo de consulta)
Unidad N° 3	<ul style="list-style-type: none"> Kernighan B, Ritchie D. (1991). El lenguaje de programación C. Pearson Education. Recuperado de https://frrq.cvg.utn.edu.ar/pluginfile.php/13741/mod_resource/content/0/El-lenguaje-de-programacion-C-2-ed-kernighan-amp-ritchie.pdf Capítulo 3 y 5 Presentaciones digitales desarrolladas por el docente. Documentación desarrollada por el docente. Banahan M, Brady D, Doran M (2019) The C book. Addison Wesley. Recuperado de https://publications.gbdirect.co.uk//c_book/ (Solo de consulta)
Unidad N° 4	<ul style="list-style-type: none"> Banahan M, Brady D, Doran M (2019) The C book. Addison Wesley. Recuperado de https://publications.gbdirect.co.uk//c_book/ Kernighan B, Ritchie D. (1991). El lenguaje de programación C. Pearson Education. Recuperado de https://frrq.cvg.utn.edu.ar/pluginfile.php/13741/mod_resource/content/0/El-lenguaje-de-programacion-C-2-ed-kernighan-amp-ritchie.pdf Capítulo 5,6 y7 Presentaciones digitales desarrolladas por el docente. Documentación desarrollada por el docente. Banahan M, Brady D, Doran M (2019) The C book. Addison Wesley. Recuperado de https://publications.gbdirect.co.uk//c_book/ (Solo de consulta)
Unidad N°5	<ul style="list-style-type: none"> Presentaciones digitales desarrolladas por el docente. Documentación desarrollada por el docente.