|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **GUÍA DE LABORATORIO DE**  **LÓGICA DE PROGRAMACIÓN** | | | | | | |
| **Unidad Didáctica: Práctica No. 2 Variables, constantes, asignaciones, estructura Secuencial** | | | | | | |
| **Eje Temático:** **Variables, constantes, asignaciones, estructura Secuencial** | | | | | | |
| **No. Guía** | | 6 | | **Resultados de Aprendizaje de la Unidad Didáctica:**  Entiende el manejo de los arreglos bidimensionales. | | |
| **1**  **sesiones** | | **15 Semana** | |
| **Horas de Trabajo** | | | |
| **Trabajo con Docente** | | **Trabajo Autónomo** | |
| **3** | | **6** | |
| **Tipo de trabajo** | | | |
| **Grupal** | **X** | **Ind** |  | **Laboratorio Requerido** | Asistido por computador / Laboratorio de Informática | |
| **Introducción** | | | | | | |
| Practicar la creación y manipulación de arreglos bidimensionales en programación, así como calcular estadísticas simples. | | | | | | |
| **Preguntas Orientadoras** | | | | | | |
| ¿Cómo crearías un arreglo en programación para almacenar las calificaciones de los estudiantes y qué tipo de datos sería apropiado usar?  ¿Qué estrategias utilizarías para calcular el promedio de las calificaciones, encontrar la calificación más alta y la más baja, y contar cuántos estudiantes obtuvieron calificaciones por encima de 90 en el arreglo? | | | | | | |
| **Presaberes Requeridos** | | | | | | |
| Se requiere conocer los fundamentos de cómo resolver un algoritmo que son: comprender, procesar, organizar, priorizar información de un problema específico utilizando métodos secuenciales para proponer una solución.   1. **Qué forma representa una “Decisión” en un Diagrama de Flujo** 2. Rectángulo 3. Rombo 4. Elipse 5. Paralelogramo 6. **¿Qué es un lenguaje de programación?** 7. Un lenguaje de programación es un conjunto de instrucciones y reglas que se utilizan para escribir programas de computadora. 8. Un lenguaje de programación es un medio de comunicación entre el programador y la computadora, que le permite escribir código y dar instrucciones al sistema. 9. Un lenguaje de programación es un conjunto de símbolos y reglas gramaticales que permiten expresar algoritmos y resolver problemas mediante la ejecución de instrucciones por parte de una computadora. 10. Un lenguaje de programación es un conjunto de herramientas y funciones predefinidas que facilitan la creación de software y permiten a los programadores desarrollar aplicaciones de manera eficiente. 11. **Qué es un algoritmo** 12. Conjunto ordenado de operaciones sistemáticas que permite hacer un cálculo y hallar la solución de un tipo de problemas. 13. Conjunto ordenado de operaciones meteorológicas que permite hacer un cálculo y hallar la solución de un tipo de problemas. 14. Conjunto ordenado de operaciones sistemáticas que permite hacer una conjetura y hallar la solución de un tipo de problemas. 15. Es una serie de instrucciones para ser ejecutadas por una computadora, que permite hacer una probabilidad y hallar la solución a algunos problemas. | | | | | | |
| **Marco conceptual o referencial \*** | | | | | | |
| Los **arreglos bidimensionales** son tablas de valores. Cada elemento de un arreglo bidimensional está simultáneamente en una fila y en una columna.  En matemáticas, a los arreglos bidimensionales se les llama [matrices](http://es.wikipedia.org/wiki/Matriz_(matem%C3%A1tica)), y son muy utilizados en problemas de Ingeniería.  En un arreglo bidimensional, cada elemento tiene una posición que se identifica mediante dos índices: el de su fila y el de su columna. Crear arreglos bidimensionales Los arreglos bidimensionales también son provistos por NumPy, por lo que debemos comenzar importando las funciones de este módulo:  **from** **numpy** **import** \*  Al igual que los arreglos de una dimensión, los arreglos bidimensionales también pueden ser creados usando la función array, pero pasando como argumentos una lista con las filas de la matriz:  a = array([[5.1, 7.4, 3.2, 9.9],  [1.9, 6.8, 4.1, 2.3],  [2.9, 6.4, 4.3, 1.4]])  Todas las filas deben ser del mismo largo, o si no ocurre un error de valor:  **>>>** array([[1], [2, 3]])  Traceback (most recent call last):  File "<stdin>", line 1, in <module>  ValueError: setting an array element with a sequence. | | | | | | |
| **Actividad de Trabajo Autónomo** | | | | | | |
| Nombre de la Actividad: "Registro de Calificaciones"  Objetivo: Practicar la creación y manipulación de arreglos en programación, así como calcular estadísticas simples.  Participantes: Personas que están aprendiendo programación y desean practicar con arreglos.  Descripción de la Actividad:  Preparación (5 minutos): El facilitador proporciona a los participantes un conjunto de calificaciones de estudiantes, por ejemplo: [85, 92, 78, 90, 88, 76, 95, 89].  Desarrollo (30-40 minutos): Los participantes trabajan individualmente o en parejas para realizar las siguientes tareas:  Crear un arreglo que contenga las calificaciones proporcionadas.  Calcular y mostrar la calificación promedio.  Encontrar y mostrar la calificación más alta y la más baja.  Contar cuántos estudiantes obtuvieron calificaciones por encima de 90 y mostrar este número.  Pruebas y Depuración (10-15 minutos): Los participantes prueban su programa con las calificaciones proporcionadas y otras calificaciones de su elección. Depuran su código si es necesario.  Compartir y Discutir (15-20 minutos): Los participantes comparten sus programas y explican cómo implementaron cada una de las tareas. Se fomenta la discusión sobre diferentes enfoques y soluciones.  Retroalimentación (10 minutos): El facilitador y otros participantes proporcionan retroalimentación sobre las implementaciones de los demás, destacando buenas prácticas y oportunidades de mejora.  Variaciones y Desafíos (Opcional): Para un desafío adicional, los participantes pueden ordenar el arreglo de calificaciones de menor a mayor o viceversa y mostrar el resultado.  Esta actividad es efectiva para fortalecer la comprensión de arreglos en programación, así como para trabajar en cálculos y estadísticas simples. Puedes adaptar la dificultad ajustando la cantidad de calificaciones o agregando desafíos adicionales, como el cálculo de la desviación estándar. | | | | | | |
| **Actividad de Comprobación del Trabajo Autónomo** | | | | | | |
| Se verificarán las respuestas a las preguntas de la sección de trabajo autónomo. | | | | | | |
| **Materiales, equipos e insumos a utilizar** | | | | | | |
| |  |  | | --- | --- | | **Materiales, equipos e insumos proporcionados por la Universidad** | | | Ítem | **Cantidad** | | Un computador por estudioso | *1* | | Conexión a Internet | *1* | | Entorno Integrado de Desarrollo (IDE) para programar en lenguaje Phyton. | *1* | | **Materiales del estudiante** | | | Ítem | **Cantidad** | | Drive o Git/GitHub | *1* | | Conexión a Internet en el caso de trabajar en la casa | *1* | | Entorno Integrado de Desarrollo (IDE) para programar en lenguaje Phyton | *1* | | | | | | | |
| **Precauciones, nivel de riesgo y recomendaciones a considerar** | | | | | | |
| |  |  |  |  | | --- | --- | --- | --- | | **CLASIFICACIÓN DEL RIESGO** | **Muy alto** | | **Medio** | | **Alto** | | **Bajo** | | **FACTORES DE RIESGO** | | **CÓMO MINIMIZAR LOS FACTORES DE RIESGO** | | | El factor de riesgo para este laboratorio está clasificado como BAJ O, debido a que la eficacia del conjunto de las medidas preventivas existentes es alta. | | En otras palabras, el riesgo está controlado | | | **RECOMENDACIONES, CONSIDERACIONES PARA EL USO DE MATERIAL Y EPP** | | | | | En todo caso se deben seguir las siguientes recomendaciones para minimizar los riesgos propios de la actividad:   * *Identificar y conocer el protocolo de seguridad de laboratorios de informática.* * *No navegar en internet sin autorización del docente.* * *No ejecutar programas sin autorización del docente.* * *No instalar en los equipos Software de ninguna índole.* * *No trasladar equipos de cómputo de su módulo sin autorización del personal del área.* * *Cuidar sus objetos personales.* * *Cada alumno tiene como responsabilidad recibir las actividades de cada clase y apropiarse del material necesario para el desarrollo de las mismas.* * *Está prohibido el ingreso o consumo de alimentos, bebidas, chicle... dentro de la sala.* * *Está prohibido el uso e ingreso de dispositivos como celulares, parlantes y memorias USB sin autorización.* * *No conectar ni desconectar dispositivos como teclados, mouse o conexiones, en caso de anomalía avisar al profesor para realizar cambios o conexiones.* * *El trabajo debe hacerse en silencio, evitando las reuniones o interrumpiendo las actividades de otros estudiantes.* * *Cuide el buen funcionamiento del equipo que la ha sido asignado, evite cambiar configuraciones o intervenir los programas y propiedades del sistema operativo, el auxiliar de laboratorio es el único autorizado.* * *Todo dispositivo (teclado o mouse) que se pierda o se dañe con intención deberá ser repuesto.* * *No portar maletines o morrales, estos deben quedar depositados en los lockers destinados para ello. En caso de duda pida el respectivo candado con los auxiliares de cada laboratorio.* * *No rayar mesas, sillas, paredes y equipos, cuidar el aseo y orden de su puesto de trabajo.*   *Se prohíbe el ingreso o exploración de páginas no autorizadas y pornográficas, es causal de sanción y expulsión (vetado) de la sala de informática por varias sesiones.* | | | | | **CONSIDERACIONES ÉTICAS** | | | | | *Tener siempre presente el código de ética del ingeniero de software en especial el ítem relacionado con:*  *Colegas: Cada ingeniero deberá apoyar y ser justos con los colegas, motivando a sus colegas sujetándose*  *al código, ayudando también a su desarrollo profesional, reconocer los trabajos de otros y abstenerse*  *a atribuirse de méritos indebidos, revisar los trabajos de manera objetiva, sincera y propiamente documentada. En caso de ser necesarias.* | | | | | | | | | | |
| **Procedimiento y Metodología de la práctica** | | | | | | |
|  | | | | | | |
| **Criterios de Entrega – Informe de Laboratorio** | | | | | | |
| * Formato de presentación PDF y adjuntar los archivos en con la extensión de la aplicación. Se debe comprimir el proyecto o proyectos que se realizaron * Uso de normas APA   + Introducción: Incluye una breve introducción a los ejercicios, explicando el propósito y los objetivos de cada ejercicio.   + Enunciado del problema: Presenta los enunciados de los problemas planteados que se resolvieron con los ejercicios, incluyendo una descripción clara y concisa de los problemas a abordar.   + Solución propuesta: Describe la solución propuesta en detalle, explicando el enfoque utilizado y los algoritmos o estructuras de datos empleados. Puedes incluir diagramas, pseudocódigo u otros elementos visuales para facilitar la comprensión de la solución.   + Implementación en lenguajes de programación a elección: Muestra el código fuente del programa en Python, asegurándote de que esté bien estructurado, documentado y sea legible. Incluye comentarios apropiados para explicar partes clave del código.   + Pruebas y resultados: Muestra evidencia de las pruebas realizadas en el programa para verificar su correcto funcionamiento. Puedes incluir capturas de pantalla, salidas de consola u otros registros relevantes.   + Análisis de resultados: Analiza los resultados obtenidos a través de las pruebas realizadas. Puedes evaluar el rendimiento del programa, la eficiencia del algoritmo utilizado o cualquier otro aspecto relevante.   + Imagen del trabajo en buena resolución de lo realizado en clase en el cuaderno borrador (IMPORTANTE y MANDATORIO).   + Conclusiones: Resume las conclusiones principales del laboratorio, resaltando los logros, las dificultades encontradas y las lecciones aprendidas durante el desarrollo del ejercicio.   + Bibliografía y referencias: Si has utilizado fuentes externas, como libros, tutoriales en línea o documentación oficial, asegúrate de citar correctamente las referencias utilizadas. | | | | | | |
| **Criterios de Evaluación – Práctica de Laboratorio** | | | | | | |
| |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | |  |  | **Puntuación** | | | | | **Criterios** | **Descripción** | **Cumple con la totalidad** | **Faltan algunos procesos o componentes** | **Falta la mayoría de los procesos o componentes** | **No presenta nada** | | Comprensión del problema | Capacidad para comprender el enunciado del problema y su contexto, identificar los requisitos y restricciones del problema, y definir un enfoque adecuado para su solución | 5 | 3 | 1 | 0 | | Implementación del código | Habilidad para escribir código limpio, bien estructurado y legible. Uso apropiado de variables, constantes, asignaciones y estructuras de control | 5 | 3 | 1 | 0 | | Resultados obtenidos | Capacidad para obtener resultados correctos y consistentes al ejecutar el programa. Evaluación de la precisión y exactitud de los resultados obtenidos | 5 | 3 | 1 | 0 | | | | | | | |
| **Palabras Clave** | | | | | | |
| PROGRAMACION ESTRUCTURADA EN PROGRAMACIÓN | | | | | | |
| **Bibliografía Recomendada** | | | | | | |
| |  |  |  | | --- | --- | --- | | **Tema** | **Subtema** | **Referente bibliográfico** | | Programación | Casos de éxito con programación | Azamar Palma, Ivan, Marco Antonio Acosta Mendizabal, Esteban de la Rosa Ramírez, and Leticia Salazar García. 2020. “APLICACIÓN MÓVIL PARA LA ENSEÑANZA DEL LENGUAJE DE PROGRAMACIÓN PYTHON. (Spanish).” Congreso Internacional de Investigacion Academia Journals 12 (2): 44–49. https://search-ebscohost-com.proxy.umb.edu.co/login.aspx?direct=true&AuthType=ip,uid&db=edb&AN=161295133&lang=es&site=eds-live&scope=site. | | Pacheco Salazar, Eleazar, Carolina Astudillo Hernández, Ricardo Flores Oliveros, Juan José Bedolla Solano, Víctor Manuel Villanueva Carmona, and Alicia Jaramillo Solís. 2018. “MAESTRO VIRTUAL INTERACTIVO APRENDIENDO C# Y JAVA. (Spanish).” Congreso Internacional de Investigacion Academia Journals 10 (7): 787–92. https://search-ebscohost-com.proxy.umb.edu.co/login.aspx?direct=true&AuthType=ip,uid&db=edb&AN=140390861&lang=es&site=eds-live&scope=site. | | Aprender con Python | Bermon Angarita, Leonardo, Amparo Prieto Taborda, Juan Diego Escobar Marquez, and Juan David Vergara Diaz. 2020. “Videojuego Para El Aprendizaje de Logica de Programacion.” Revista Educacion En Ingenieria 16 (31): 46. doi:10.26507/rei.v16n31.1141. | | **LIBROS Y RECURSOS DIGITALES**  Arias, Á. (2015). Aprende a Programar en Python: 2º Edición. IT Campus Academy. <https://books.google.es/books?id=oO_LCgAAQBAJ&lpg=PA9&ots=WEuf9PFSCq&dq=aprender%20python&lr&hl=es&pg=PA9#v=onepage&q=aprender%20python&f=false>  Marzal Varó, A., & Gracia Luengo, I. (2009). Introducción a la programación con Python. Universitat Jaume I. <https://repositori.uji.es/xmlui/bitstream/handle/10234/24305/s23.pdf?sequence=6>  Walker, J. S. (2018). Python: La Guía Definitiva para Principiantes para Dominar Python. Babelcube Inc.. https://books.google.es/books?id=T15tDwAAQBAJ&lpg=PT6&ots=EQUoRIs9cU&dq=aprender%20python&lr&hl=es&pg=PT6#v=onepage&q=aprender%20python&f=false | | | | | | | |
| **Control de cambios** | | | | | | |
| **Fecha de Actualización** | | | | **Descripción** | | **Participantes** |
| 24/10/2023 | | | |  | | Carlos Alberto Cañón Romero |