25-nov-2020 07:28:00

PWI\_006\_007 AVAC

|  |  |  |  |
| --- | --- | --- | --- |
| **FORMATO DE GUÍA DE PRÁCTICA DE LABORATORIO / TALLERES / CENTROS DE SIMULACIÓN – PARA DOCENTES** | | | |
| **CARRERA**: CIENCIAS DE LA COMPUTACIÓN | | | **ASIGNATURA: ESTRUCTURA DE DATOS** |
| **NRO. PRÁCTICA**: | --- | **TÍTULO** : PROYECTO INTEGRADOR 2DO BIMESTRE | |
| **INTEGRANTE(S):** | | **Ricardo Maigua, Jefferson Cajas, Stalin Cacuango, Carlos Prado** | |
| **OBJETIVO DEL PROYECTO:**   * Desarrollar una aplicación de escritorio funcional utilizando el paradigma orientado a objetos. * Implementar estructuras de datos al proyecto con el fin de brindar una mejor funcionalidad. | | | |
| **ACTIVIDADES POR DESARROLLAR** | | | |
| **DESCRIPCIÓN DEL PROYECTO:**  Se requiere una aplicación de escritorio que permita registrar las notas de estudiantes universitarios, del primer y segundo parcial sobre 50 puntos. La aplicación está enfocada al docente. Como entrada se tiene el listado de estudiantes y sus notas. Como salida, el programa proporciona una gráfica estadística y un análisis de las notas. La aplicación debe ser funcional, con interfaces intuitivas y manejar correctamente los datos.  **Requerimientos funcionales**  El ingreso de información consta de dos partes: En primer lugar, el usuario ingresa el listado de estudiantes de su asignatura. Se debe registrar los apellidos y nombres de cada uno. Luego, debe existir una interfaz que permita ingresar las notas de cada alumno. Tanto los nombres, como las notas se podrán modificar, sin perder la información previamente ingresada. Cabe destacar, que deben ser interfaces distintas para el ingreso del listado y para el ingreso de notas.  El programa consta de otras dos interfaces que permiten mostrar los siguientes resultados. Primero, se visualiza una gráfica en barras que muestra las notas finales (suma de las notas de ambos parciales) por cada estudiante y una línea que representa el promedio del curso (ambas gráficas sobre la misma imagen). Si el puntaje alcanzado por cada estudiante es inferior a 70 puntos, la barra se debe pintar de color rojo.  Por otro lado, el programa debe generar un árbol binario de búsqueda considerando la nota final de los estudiantes. El análisis inicia tomando como nodo raíz la nota promedio del curso. Cada nuevo nodo (que representa a un alumno) se debe ubicar según corresponda dentro del árbol, dependiendo de la nota alcanzada por el estudiante. El árbol debe visualizar el nombre del estudiante y la nota final.  Finalmente, la aplicación debe almacenar toda la información en archivos los datos de los estudiantes (listado, notas y gráficas).  **Requerimientos no funcionales**  El diseño y desarrollo de la aplicación debe realizarse considerando los siguientes requisitos:  Estructura de Datos:   * Utilizar al menos dos estructuras de datos diferentes para manejar la información de la aplicación (Colas, Pilas, Listas, árboles, etc) * Aplicar correctamente árboles binarios de búsqueda para el análisis de notas   Programación Orientada a Objetos   * Aplicar todos los conceptos de programación orientada a objetos. Los alumnos deben ser manipulados como objetos dentro de la aplicación. * Utilizar un diagrama de clases para representar toda la aplicación (utilizar asociaciones). * Programar utilizando el patrón MVC. * Aplicar conceptos de usabilidad sobre las interfaces gráficas. * Almacenar la información de los alumnos utilizando archivos de texto CSV.  1. **Desarrollo de la práctica (Uso del algoritmo, pruebas de escritorio, ejecución, captura de pantallas).**          1. **Recursos utilizados (Computadora, libros, proyector, etc.)**     Cada uno de nosotros utilizamos nuestra laptop además de haber usado QT creator para la programación del algoritmo de este proyecto, y también usamos algunos códigos hechos en qt para guiarnos , y por ultimo una conexión a internet para tener información sobre el marco teórico. | | | |
| **RESULTADO(S) OBTENIDO(S)**:  Se pudo lograr con éxito el programa se puede ingresar notas de los estudiantes de su respectiva asignatura y se puede observar su promedio y el promedio total del curso además de verse en una grafica de barras se puede observar su árbol de notas impreso gráficamente como se pudo observar en las capturas de pantallas. | | | |
| **CONCLUSIONES**:   * Se pudo identificar las características y generalidades de las técnicas para realizar programación dinámica. * Aprendimos a crear interfaces gráficas de usuario empleando técnicas de modelado de objetos. * Ahora manejamos con facilidad la implementación de clases, ingreso y almacenamiento de datos. | | | |
| **RECOMENDACIONES**:  Como recomendaciones podemos decir que fue muy difícil realizar este proyecto más la parte de los árboles en qt ya que fue muy complejo graficar correctamente estos, debido que no había ninguna librería ni nada que nos ayudara a implementar nudos de estos arboles y esta fue la parte más difícil pero todo se pudo resolver con éxito. | | | |

***Msc. Walter Gaibor*:**

# *Firma*: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_