Interfaz de usuario gráfica, Texto

Descripción generada automáticamente

**Paradigmas de Programación**

**Trabajo**

*La taxonomía de Bloom*

**INTEGRANTES**

Martín Araya Díaz

Carlos Gonzales

Ignacio Benegas

**Introducción**

Evaluar el aprendizaje en programación es un desafío constante, sobre todo si queremos ir más allá de que los estudiantes memoricen conceptos. La Taxonomía de Bloom nos ayuda a estructurar preguntas según distintos niveles cognitivos, como 'Recordar', 'Aplicar' o 'Analizar'. En este trabajo, diseñamos un programa en C++ que permite crear un banco de preguntas y generar evaluaciones adaptadas a esos niveles.

**Descripción de la Solución**

El programa está compuesto por tres clases principales:

1. Pregunta: representa una pregunta con atributos como enunciado, tipo (VF o OM), respuesta, nivel taxonómico, año y tiempo.

2. BancoPreguntas: permite agregar, eliminar, actualizar y buscar preguntas.

3. Evaluacion: agrupa un conjunto de preguntas seleccionadas para una prueba, calcula el tiempo total estimado.

**Ejemplo de Clase Pregunta**

Captura de pantalla de computadora

Descripción generada automáticamente

**Ejemplo de Menú en main()**

**Texto

Descripción generada automáticamente**

**Bibliotecas utilizadas**

vector: Se usa para almacenar dinámicamente las preguntas en las clases BancoPreguntas y Evaluacion. Un std::vector se usa para manejar colecciones de objetos debido a su flexibilidad y eficiencia.

string: Se usa para manejar cadenas de texto, como el enunciado, tipo, respuesta y nivel taxonómico de las preguntas. std::string permite una manipulación eficiente de las cadenas.

iostream: Se usa para la entrada y salida estándar del programa, asi funciona la interacción con el usuario a través de std::cin y std::cout.

algorithm: Se usa en la función eliminarPregunta de la clase BancoPreguntas para reorganizar el vector de preguntas y eliminar las preguntas deseadas.

limits: Se usa en la función leerEntero para manejar entradas inválidas y evitar errores de entrada de datos.

**Conclusión**

El sistema funciona correctamente para crear y gestionar un banco de preguntas, clasificadas por nivel cognitivo. La generación de evaluaciones personalizadas y la estimación del tiempo total ayudan a crear pruebas balanceadas. En el futuro, podríamos agregar guardado en archivos o interfaz gráfica. Lo más valioso fue aplicar la teoría de Bloom a un problema real usando programación orientada a objetos.