



XVIII
Olimpiada
de Informática
y Tecnología
Colegio Bilingüe José Max León

Torneo de Programación

Reto Programación Abierta (6° a 11°)

"Acceso Sin Barreras: Información Escolar para Todos + Análisis de Uso"

Introducción:

Bienvenidos al Colegio Bilingüe José Max León. Aquí tendrás la oportunidad de demostrar tus habilidades para programar así resolver el reto propuesto. Recuerda que lo más importante es que tu creatividad y pensamiento lógico sean puestos en práctica y que tú te sientas a gusto y orgulloso de hacer parte de esta increíble jornada.

CONTEXTO

El valor de la inclusión

La inclusión implica que todas las personas sean valoradas y respetadas, sin importar sus diferencias. Por ejemplo, en un colegio en el que la inclusión es parte importante del día a día, cada estudiante tiene la oportunidad de aprender, participar y expresarse libremente, sintiéndose seguro y aceptado.

Practicar la inclusión nos ayuda a construir relaciones basadas en la empatía, el respeto y la solidaridad y nos permite reconocer que la diversidad nos enriquece como comunidad.

Cuando promovemos la inclusión, contribuimos a prevenir el acoso escolar y a crear un ambiente donde todos pueden crecer y aprender juntos.

OBJETIVOS

- Integrar diferentes algoritmos
- Crear y emplear diferentes variables y procesos
- Crear una interacción entre usuario y el software, con elementos de inclusión.
- Incluir condiciones y generar una interacción, que le permita a cualquier usuario participar dentro del programa



CONTEXTO DEL PROBLEMA¹

En los colegios, muchos procesos están disponibles digitalmente, pero diseñados para usuarios con habilidades estándar. Procesos tales como:

-  Circulares
-  Horarios
-  Eventos institucionales (incluida la XVIII Olimpiada Intercolegiada)
-  Informe académico trimestral

Esto puede generar barreras para:

-  Adultos mayores con baja visión.
-  Padres con discapacidad auditiva.
-  Personas con dificultades visuales.
-  Estudiantes en proceso de lectura.

La inclusión no solo implica permitir el acceso, sino también analizar cómo se usa la información para mejorar el servicio.

PROBLEMA ESPECÍFICO

Diseñar y programar una aplicación accesible que permita a un usuario con condición de inclusión acceder a mínimo **dos** módulos escolares, incluyendo obligatoriamente **uno** de los siguientes:

- Información del evento de la Olimpiada Intercolegiada.
- Informe académico trimestral simulado.

Además, la aplicación deberá incorporar un sistema básico de **registro estadístico** de consultas.

¹ El diseño metodológico del presente reto contó con apoyo de inteligencia artificial generativa (ChatGPT, OpenAI, 2026), como herramienta de estructuración académica. La validación pedagógica y contextualización institucional fue realizada por el equipo docente del área de Tecnología.



XVIII
Olimpiada
de Informática
y Tecnología
Colegio Bilingüe José Max León

REQUISITOS TÉCNICOS

Lenguajes permitidos:

- Python (Preferible)
- Java
- C++
- JavaScript

Tipo de aplicación: Puede ser una de las siguientes

- Consola accesible
 - Interfaz gráfica
-



CONDICIONES OBLIGATORIAS

1 Usuario claramente definido

Ejemplo:

“Abuela con baja visión que consulta el informe académico de su nieto.”

2 Acceso funcional a dos módulos

Debe permitir consultar información simulada como:

- Fecha, lugar y descripción de la Olimpiada.
- Notas por asignatura del informe trimestral.
- Circular institucional.
- Horario escolar.



3 Implementación mínima de tres principios de accesibilidad

Ejemplos:

- Ajuste de tamaño de texto.
- Alto contraste.
- Navegación simplificada.
- Lectura por voz.
- Alertas visuales.
- Lenguaje simplificado.
- Subtítulos.

Los principios pueden fundamentarse en las recomendaciones del World Wide Web Consortium (WCAG 2.1), la Organización Mundial de la Salud o la Organización de las Naciones Unidas.

4 Registro Estadístico Obligatorio

La aplicación debe incluir un sistema que registre como mínimo:

- Número total de consultas realizadas.
- Cantidad de consultas por módulo.
- Usuario simulado que realizó la consulta.
- Fecha u orden de consulta.

Debe permitir:

- Mostrar estadísticas generales.
- Identificar cuál módulo es más consultado.
- Generar al menos una conclusión básica a partir de los datos.

Ejemplo de funcionalidad esperada:

“El módulo de informe académico fue consultado 12 veces, siendo el más utilizado por usuarios con baja visión.”

Se evaluará:

- Uso de estructuras de datos (listas, diccionarios, arreglos, clases).
 - Lógica de conteo y almacenamiento.
 - Claridad en la visualización de resultados.
-



XVIII
Olimpiada
de Informática
y Tecnología
Colegio Bilingüe José Max León

Competencias Evaluadas

- Programación estructurada.
 - Diseño centrado en el usuario.
 - Aplicación de principios de accesibilidad.
 - Análisis básico de datos.
 - Impacto social institucional.
-

Sustentación (Máximo 10 minutos)

- **Tipo de evaluación:** Sustentación individual.
 - **Fecha:** 25 de marzo
 - **Tiempo de sustentación:** 10 minutos
 - **Evidencias:**
 - Programa funcional.
 - Código fuente.
 - Opcional: Evidencias adicionales que considere necesarias, ejemplo: Testimonios, datos de aplicación, etc.
 - Opcional: Presentación del proyecto (Powerpoint, Googlesheets, Canva, etc).
 - **Estructura sugerida para la sustentación:**
 - Usuario y barrera identificada (2 min)
 - Problema institucional (2 min)
 - Explicación técnica del desarrollo (2 min)
 - Demostración funcional de los módulos y estadísticas (3 min)
 - Conclusión sobre impacto institucional (1 min)
-



RÚBRICA DE EVALUACIÓN

Categoría 3. Programación Estructurada Python, Java, C++ (6° a 11°)

Colegio

Nombre de los estudiantes

Criterio	Bajo 4 Puntos	En desarrollo 8 Puntos	Básico 12 Puntos	Alto 14 Puntos	Superior 16 Puntos
1. Pertinencia del usuario y problema	Poco claro	Definición limitada	Claro	Bien contextualizado	Profundo análisis institucional
2. Acceso funcional a los dos módulos	No funciona	Funciona parcialmente	Funciona con limitaciones	Funciona correctamente	Funciona de manera intuitiva y centrada en el usuario
3. Implementación de accesibilidad	No aplica criterios	Aplica 1 criterio	Aplica 2 criterios	Aplica 3 criterios coherentes	Integra múltiples principios alineados con WCAG
4. Registro estadístico y análisis	No registra datos	Registra datos sin análisis	Registra y muestra conteos básicos	Presenta estadísticas claras	Presenta análisis estructurado y conclusiones relevantes
5. Calidad técnica del código	Desorganizado	Funcional limitado	Organizado básico	Buen uso de estructuras y modularidad	Código limpio, eficiente, bien documentado y escalable
6. Sustentación oral	Desorganizada	Poco clara	Clara pero limitada	Clara y estructurada	Argumentación sólida, dominio técnico y excelente manejo del tiempo

PUNTOS ADICIONALES DE EXCELENCIA (0–4 puntos)

Criterio otorgado únicamente cuando el proyecto demuestra un nivel excepcional integral:

- ☐ 0 puntos – No aplica
- ☐ 2 puntos – Destacado
- ☐ 4 puntos – Sobresaliente institucional



TOTAL FINAL

Suma de los 6 criterios (máximo 96):

Puntos de excelencia (máximo 4):



PUNTAJE

/100

FINAL:



Observaciones del jurado

Fortalezas principales:

Aspectos por mejorar:



Mensaje Final para los Participantes

No basta con permitir el acceso.

Una tecnología verdaderamente inclusiva también aprende de cómo las personas la usan para mejorar continuamente.

En esta Olimpiada, el reto no es solo programar...

Es diseñar una solución que incluya, funcione y analice para mejorar.



Modelo de Estructura de Datos sugerido para orientar a los docentes acompañantes.

⚠ Este modelo **no es obligatorio**, pero servirá como referencia técnica para garantizar profundidad académica, orden estructural y coherencia en el desarrollo.



MODELO DE ESTRUCTURA DE DATOS SUGERIDO

Reto: “Acceso Sin Barreras – 2026”

1 Modelo Conceptual General

La aplicación puede estructurarse en **4 componentes principales**:

1. Gestión de Usuarios
2. Gestión de Módulos Institucionales
3. Sistema de Consultas
4. Sistema Estadístico

2 MODELO ORIENTADO A OBJETOS (Recomendado para nivel avanzado)



Clase Usuario

Atributos sugeridos:

- id
- nombre
- edad
- tipoNecesidad (baja_vision, auditiva, lectura_inicial, etc.)
- nivelAccesibilidadPreferido (alto_contraste, voz, texto_simple)

Métodos sugeridos:

- mostrarPerfil()
 - seleccionarModulo()
-



XVIII
Olimpiada
de Informática
y Tecnología
Colegio Bilingüe José Max León

Clase Módulo

Puede ser clase base.

Atributos:

- idModulo
- nombreModulo
- descripcion
- tipo (evento, informe, circular, horario)

Métodos:

- mostrarInformacion()
 - mostrarVersionAccesible()
-

Subclases de Módulo

- Evento
- InformeAcademico
- Circular
- Horario

Ejemplo adicional para InformeAcademico:

- estudiante
 - materias (estructura tipo diccionario o lista de objetos)
 - promedio
-

Clase Consulta

Atributos:

- usuario
 - modulo
 - fechaConsulta
 - numeroConsulta
-

Clase Sistema Estadistico

Atributos:

- totalConsultas
- consultasPorModulo (diccionario)
- consultasPorUsuario (diccionario)



XVIII
Olimpiada
de Informática
y Tecnología
Colegio Bilingüe José Max León

Métodos:

- registrarConsulta()
- mostrarEstadisticasGenerales()
- obtenerModuloMasConsultado()
- obtenerUsuarioMasActivo()

3 MODELO ESTRUCTURADO (Para nivel Junior o consola)

Si no se usa programación orientada a objetos, se sugiere:

Diccionario Principal del Sistema

Ejemplo conceptual:

```
sistema = {  
  "usuarios": [],  
  "modulos": {},  
  "consultas": [],  
  "estadisticas": {  
    "total": 0,  
    "por_modulo": {},  
    "por_usuario": {}  
  }  
}
```

Estructura de Usuarios

```
usuarios = [  
  {  
    "id": 1,  
    "nombre": "María Rodríguez",  
    "edad": 68,  
    "necesidad": "baja_vision"  
  }  
]
```



XVIII
Olimpiada
de Informática
y Tecnología
Colegio Bilingüe José Max León



Estructura de Informe Académico

```
informe = {  
  "estudiante": "Sofía Martínez",  
  "curso": "8B",  
  "materias": {  
    "Matemáticas": 4.3,  
    "Inglés": 4.7,  
    "Tecnología": 4.9  
  },  
  "promedio": 4.5  
}
```



Registro de Consulta

```
consulta = {  
  "usuario": "María Rodríguez",  
  "modulo": "Informe Académico",  
  "numero": 5  
}
```

4 MODELO DE BASE DE DATOS SIMULADA (Opcional Nivel Avanzado)

Para equipos que deseen mayor complejidad:

Tablas sugeridas:

Usuarios (id, nombre, edad, necesidad)

Modulos (id, nombre, tipo)

Consultas (id, idUsuario, idModulo, fecha)

Informes (id, estudiante, curso, promedio)

Se valorará uso de:

- Archivos JSON
- Archivos CSV
- Persistencia básica
- Serialización



5 LÓGICA MÍNIMA ESPERADA

El sistema debería permitir:

1. Seleccionar usuario.
2. Mostrar menú accesible.
3. Elegir módulo.
4. Mostrar información adaptada.
5. Registrar automáticamente la consulta.
6. Actualizar estadísticas.
7. Mostrar resumen estadístico.

6 NIVEL TÉCNICO DIFERENCIADO

◆ Nivel Junior

- Uso correcto de listas y diccionarios.
- Funciones modulares.
- Registro estadístico básico.
- Validación de entradas.

◆ Nivel Avanzado

- Uso obligatorio de clases.
- Herencia o composición.
- Separación lógica en módulos.
- Manejo de errores.
- Análisis estadístico adicional (porcentajes, ranking).

7 Recomendaciones para Docentes Acompañantes

Orienta a los estudiantes en:

- Diseñar primero el modelo de datos antes de programar.
- Identificar claramente qué información se almacena y por qué.
- Separar lógica de presentación.
- Garantizar que el registro estadístico no sea superficial.
- Evitar soluciones lineales sin modularidad.



Enfoque Pedagógico Clave

Este reto no solo evalúa programación.

Evalúa:

- Diseño estructurado.
- Pensamiento sistémico.
- Responsabilidad social en tecnología.
- Capacidad de análisis de datos.