

Mileva II

Modelado didáctico de interacción humano-computadora

Presentación

Mileva II es una aplicación web interactiva orientada a la enseñanza inicial de la programación. Propone actividades breves en las que el estudiante completa o analiza instrucciones, las ejecuta y observa de inmediato los cambios producidos en variables, objetos y representaciones visuales. El recurso busca reducir la abstracción inicial y favorecer la comprensión del estado de un programa, la secuenciación y la lógica booleana.

Origen y continuidad del proyecto

Mileva tiene su origen en el proyecto de grado “**Modelado didáctico de interacción humano-computadora**”, desarrollado en la Facultad de Ingeniería de la Universidad de la República por Eliana Rosselli Orrico, Santiago Correa y Cecilia Guayta, bajo la supervisión de Sylvia da Rosa y Federico Gómez Frois. El trabajo fue presentado como tesis de Ingeniería en Computación en 2023 y dio lugar a una herramienta concebida específicamente para apoyar la construcción de conceptos básicos de programación.

Mileva II continúa esa línea de desarrollo mediante un recorrido renovado, organizado en dos grandes módulos: variables y asignaciones, y lógica booleana. La propuesta mantiene como eje la interacción entre instrucciones, estado interno y representación visual.

Agradecimiento. Desde Uruguay Educa se agradece especialmente al equipo de la FING-Udelar la disposición para compartir Mileva II y hacer posible su difusión como recurso educativo. Esta colaboración amplía el acceso de docentes y estudiantes a una propuesta universitaria de calidad y con una clara finalidad didáctica.



Figura 1. Pantalla principal y organización modular de Mileva II. Fuente: captura del recurso.

Una interacción pensada para aprender

El diseño combina una zona de instrucciones con una representación visual del problema. El estudiante puede anticipar un resultado, completar una expresión, ejecutar la solución, observar sus efectos, identificar errores y reiniciar la actividad. La retroalimentación no se limita a indicar si una respuesta es correcta: permite ver cómo cada instrucción modifica el escenario.

Progresión guiada Los ejercicios avanzan desde asignaciones simples hacia expresiones y combinaciones lógicas.	Estado visible Los cambios en variables se traducen en movimientos, luces, accesos, puertas u otros estados gráficos.
Carga sintáctica reducida La atención se concentra en el significado de las instrucciones antes de abordar un lenguaje completo.	Contextos variados Una misma idea se presenta mediante problemas diferentes, favoreciendo la transferencia conceptual.
Ensayo y depuración Los controles de ejecutar y reiniciar promueven la formulación de hipótesis y la revisión de errores.	Puente hacia lo textual Las actividades pueden luego traducirse a pseudocódigo, bloques, Python u otros lenguajes.

Ejercicio 1

Presioná 'Ejecutar' para ver cómo se determina si la persona puede entrar al cine. La variable **puedeEntrar** es booleana: queda en **true** cuando la comparación **edad >= 16** es verdadera (al menos 16 años) y queda en **false** en caso contrario. Variables disponibles: **edad**, **puedeEntrar**.

1. edad = 18

2. puedeEntrar = edad >= 16

Ejecutar

Resetear



Figura 2. Una comparación relacional produce un valor booleano y modifica la representación visual.

Ejercicio 11

Para entrar al laboratorio, necesito tener: la llave y la bata de seguridad puesta. Completá la expresión para que **puedeEntrar** sea true solamente cuando tenga ambas cosas. Recordatorio: **puedeEntrar = tengoLlave AND tengoBata**. Variables disponibles: **tengoLlave**, **tengoBata**, **puedeEntrar**.

1. tengoLlave = true

2. tengoBata = true

3. puedeEntrar =

Ejecutar

Resetear

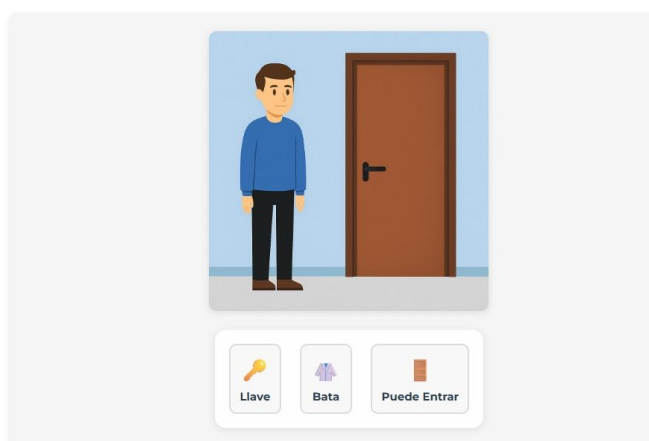


Figura 3. Los operadores booleanos se trabajan mediante condiciones cercanas y observables.

Contenidos y progresión

La portada organiza el recorrido en diez secciones. El primer módulo aborda el funcionamiento de las variables y la secuencia de instrucciones; el segundo introduce valores booleanos, comparaciones y operadores lógicos. Esta organización permite utilizar el recurso de forma lineal o seleccionar actividades puntuales según las necesidades del grupo.

Módulo	Recorrido	Conceptos centrales
1. Variables y asignaciones	Tutorial; asignación; orden de instrucciones; intercambio de valores; introducción a la formalización.	Variable, valor, estado, secuencia, actualización e intercambio.
2. Lógica booleana	Variables booleanas; expresiones; operadores; evaluación por circuito corto; formalización.	true/false, comparación, AND, OR, NOT, precedencia y circuito corto.

Ejercicio 1

Tutorial: circuito corto con AND. Observá que cuando **luzEncendida1** es **false**, el operador AND no necesita evaluar el segundo operando (**luzEncendida2**) porque ya sabe que el resultado será **false**. Esto se llama **evaluación por circuito corto**.

Variables disponibles: *ninguna* (no se les puede cambiar el valor directamente: **luzEncendida1**, **luzEncendida2**, **luzTotal**).

1.
2.
3.

Ejecutar
Resetear

1



2



Figura 4. El tutorial hace visible que una expresión AND puede resolverse sin evaluar el segundo operando cuando el primero es false.

Ejercicio 5

Entrada al laboratorio: completá la expresión para habilitar el acceso si hay modo de emergencia o si alguna puerta está abierta sin alarma.

Variables disponibles: **puertaAbierta1**, **puertaAbierta2**, **puertaAbierta3**, **puertaAbierta4**, **puertaAbierta5**, **alarma1**, **alarma2**, **alarma3**, **alarma4**, **alarma5**, **modoEmergencia** (no se les puede cambiar el valor directamente: **accesoPermitido**).

1. 

```
accesoPermitido = modoEmergencia OR
((puertaAbierta1 AND NOT(alarma1)) OR
(puertaAbierta2 AND NOT(alarma2)) OR
(puertaAbierta3 AND NOT(alarma3)) OR
(puertaAbierta4 AND NOT(alarma4)) OR
(puertaAbierta5 AND NOT(alarma5)))
```

2.

Ejecutar
Resetear

1



2



3



4



5



Figura 5. Las situaciones avanzan hacia expresiones compuestas con varios operadores y condiciones.

La secuencia no se limita a reconocer símbolos: propone interpretar condiciones, relacionarlas con el estado del escenario y formalizarlas mediante expresiones. De este modo, el estudiante puede avanzar desde una comprensión intuitiva hacia una representación cada vez más precisa.

Valor pedagógico y orientaciones de uso

- **Construcción de modelos mentales.** la correspondencia inmediata entre instrucción y resultado ayuda a comprender que un programa mantiene un estado que cambia durante la ejecución.
- **Predicción, prueba y depuración.** las actividades permiten anticipar resultados, contrastarlos y revisar la solución, prácticas centrales del pensamiento computacional.
- **Foco conceptual.** al reducir la complejidad sintáctica, Mileva II concentra la atención en variables, secuencias, comparaciones y lógica.
- **Transferencia.** los problemas pueden reconstruirse después en pseudocódigo, programación por bloques o lenguajes textuales de propósito general.

Articulación con Ciencias de la Computación

El recurso presenta una correspondencia especialmente directa con los contenidos de Algoritmia y programación de 7.º y 8.º de Educación Básica Integrada: variables, asignaciones, orden de ejecución, expresiones lógicas, decisiones, prueba y depuración. En 9.º puede utilizarse como diagnóstico, recuperación o nivelación antes de avanzar hacia programas textuales con estructuras y abstracciones de mayor complejidad.

Alcance. Mileva II no sustituye una secuencia completa de enseñanza. Su mayor potencial aparece cuando se combina con instancias de explicación, registro de trazas, comparación de soluciones, traducción a otros lenguajes y producción de problemas propios.

Continuidad con Mileva I

La primera versión de Mileva ya se encuentra publicada en Uruguay Educa como [Mileva: modelado didáctico de interacción humano-computadora](#). La publicación de Mileva II permite dar continuidad a ese vínculo entre la producción universitaria y la educación media, ofreciendo una versión renovada del recurso y ampliando sus posibilidades de uso en el aula.

Sugerencias para integrarla en el aula

1. Solicitar una predicción antes de ejecutar y pedir que se justifique qué variable o condición cambiará.
2. Registrar una traza breve con la instrucción, el valor anterior y el valor posterior.
3. Comparar soluciones diferentes y discutir si producen o no el mismo resultado.
4. Cerrar con una transferencia: reescribir una actividad en pseudocódigo, bloques o un lenguaje textual.

Accesos y reconocimiento

Mileva II: [acceder al recurso](#) | **Proyecto de grado en Colibrí:** [consultar la tesis](#) | **Contacto publicado:** mileva-fing@googlegroups.com

Uruguay Educa reconoce y agradece el trabajo del equipo que dio origen a Mileva y la generosidad de compartir Mileva II para su difusión. La incorporación del recurso al portal contribuye a poner en circulación una propuesta abierta, visual y pedagógicamente valiosa para la enseñanza de la programación.

Referencia del proyecto: Rosselli Orrico, E., Correa, S. y Guayta, C. (2023). Modelado didáctico de interacción humano-computadora [Tesis de grado, Universidad de la República, Facultad de Ingeniería, INCO].