



A|N|I|I

AGENCIA NACIONAL
DE INVESTIGACIÓN
E INNOVACIÓN



FACULTAD DE
CIENCIAS
UDELAR fcien.edu.uy



UNIVERSIDAD
DE LA REPÚBLICA
URUGUAY

Integración del pensamiento computacional en la enseñanza de la cinemática

Mateo Dutra Shaw

Programa de Desarrollo de las Ciencias Básicas - Física
Facultad de Ciencias
Universidad de la República

Montevideo – Uruguay
Diciembre de 2025



A|N|I|I

AGENCIA NACIONAL
DE INVESTIGACIÓN
E INNOVACIÓN



FACULTAD DE
CIENCIAS
UDELAR fcien.edu.uy



UNIVERSIDAD
DE LA REPÚBLICA
URUGUAY

Integración del pensamiento computacional en la enseñanza de la cinemática

Mateo Dutra Shaw

Tesis de Maestría presentada al Programa de Desarrollo de las Ciencias Básicas - Física y la Facultad de Ciencias de la Universidad de la República, como parte de los requisitos necesarios para la obtención del título de Magíster en Física.

Director:

Ph.D. Arturo Martí

Codirector:

Ph.D. Álvaro Suárez

Montevideo – Uruguay

Diciembre de 2025

Dutra Shaw, Mateo

Integración del pensamiento computacional en la enseñanza de la cinemática / Mateo Dutra Shaw. - Montevideo: Universidad de la República, Facultad de Ciencias, 2025.

[IX, 103](#) p. 29, 7cm.

Director:

Arturo Martí

Codirector:

Álvaro Suárez

Tesis de Maestría – Universidad de la República, Programa de Desarrollo de las Ciencias Básicas - Física, 2025.

Referencias bibliográficas: p. [94 – 100](#).

1. Pensamiento Computacional,
 2. Cinemática,
 3. Interpretación de gráficas,
 4. Python,
 5. PER.
- I. Martí, Arturo, *et al.* II. Universidad de la República, Programa de Desarrollo de las Ciencias Básicas - Física III. Título.

INTEGRANTES DEL TRIBUNAL DE DEFENSA DE TESIS

Ph.D. Marcela Peláez

Ph.D. Kristina Zuza

Ph.D. Sylvia Da Rosa

Ph.D. Cecilia Stari

Ph.D. Lucía Amy

Montevideo – Uruguay
Diciembre de 2025

Agradecimientos

Quiero comenzar agradeciendo a mis tutores, Arturo y Álvaro, por el excelente trabajo de orientación que realizaron a lo largo de este proceso. No solo por su disposición constante, las sugerencias, correcciones y perspectivas compartidas, sino también porque fueron piezas fundamentales para que este trabajo fuera posible. A Arturo, por abrirme las puertas y confiar en mí para integrarme por primera vez a proyectos de investigación. A Álvaro, por haber realizado la primera maestría y el primer doctorado en PER del país, mostrando el valor de las investigaciones en el área y allanando el camino para quienes llegamos después. También quiero agradecer a Cecilia y Martín por ser parte desde mis primeros pasos de mi recorrido en la investigación en enseñanza de la física.

Les agradezco a todos los que forman o formaron parte del 326, 330, 333 y la 20, con quienes compartí el día a día durante estos años de trabajo en la maestría. En especial a Juanpe, un gran amigo con quien compartimos todas las etapas de formación en física. También quiero agradecer a mis compañeros del liceo donde doy clases, que fueron fundamentales desde mis primeros pasos como docente.

Extiendo mi agradecimiento al tribunal por aceptar la invitación y dedicar su tiempo a la lectura y corrección de este trabajo; a la ANII, por el apoyo económico que lo hizo posible; y a Jimena, por su excelente gestión, que nos facilita enormemente el camino a los estudiantes.

Finalmente, a mi familia y amigos de siempre por el apoyo incondicional y acompañamiento durante estos años. Y a Agustina, por la motivación que me brindó en los últimos meses para que pudiera terminar esta etapa.

RESUMEN

El uso de recursos computacionales juega un papel central en la investigación en todos los campos de la física. Sin embargo, en la enseñanza de la disciplina su integración aún plantea importantes desafíos, especialmente en los niveles iniciales. Esta tesis explora la incorporación del Pensamiento Computacional (PC) en la enseñanza de la cinemática, a partir del diseño, implementación y evaluación de una secuencia didáctica desarrollada en un curso de educación media. El enfoque teórico adoptado permitió vincular los procesos de construcción del conocimiento con las prácticas de PC, orientando tanto el diseño de las actividades como el análisis de las producciones de los estudiantes. La secuencia incluyó tareas de elaboración de algoritmos, simulaciones y análisis de datos en Python, articuladas con la enseñanza de los conceptos de posición, velocidad y aceleración.

La evaluación de la secuencia combinó estrategias cualitativas y cuantitativas orientadas a caracterizar el desarrollo de las prácticas de PC y su articulación con los aprendizajes en cinemática. Los resultados mostraron que el trabajo con algoritmos, simulaciones y análisis de datos permitió a los estudiantes vincular expresiones matemáticas, procedimientos de cálculo y representaciones gráficas de un modo más integrado, favoreciendo la comprensión de los conceptos de movimiento. En particular, las actividades que involucraron la descomposición de problemas, la traducción de relaciones físicas a código y la construcción de tablas y gráficos a partir de los resultados del programa promovieron una exploración más profunda de las relaciones entre posición, velocidad y tiempo, que difícilmenteemergería en un enfoque tradicional.

En conjunto, los hallazgos indican que la incorporación del PC en la enseñanza de la física es viable y aporta un valor formativo específico, al abrir nuevas vías para pensar los fenómenos de movimiento a través de la programación. Como líneas de continuidad, se plantea fortalecer la integración entre física y programación, diseñar nuevas formas de evaluar los aprendizajes que emergen de esa articulación y desarrollar recursos que faciliten su implementación en el aula.

Palabras claves:

Pensamiento Computacional, Cinemática, Interpretación de gráficas, Python, PER.

ABSTRACT

The use of computational resources plays a central role in research across all fields of physics. However, their integration into the teaching of the discipline still presents significant challenges, especially at the introductory levels. This thesis explores the incorporation of Computational Thinking (CT) into the teaching of kinematics through the design, implementation, and evaluation of an instructional sequence developed in a secondary-level physics course. The theoretical framework adopted made it possible to connect knowledge construction processes with CT practices, guiding both the design of the activities and the analysis of students' productions. The sequence included tasks involving algorithm development, simulations, and data analysis in Python, articulated with the teaching of the concepts of position, velocity, and acceleration.

The evaluation combined qualitative and quantitative strategies aimed at characterizing the development of CT practices and their relationship with students' learning in kinematics. The results showed that working with algorithms, simulations, and data analysis enabled students to link mathematical expressions, calculation procedures, and graphical representations in a more integrated way, which favored their understanding of motion concepts. In particular, activities involving problem decomposition, the translation of physical relationships into code, and the construction of tables and graphs from program outputs fostered a deeper exploration of the relationships among position, velocity, and time—an approach unlikely to emerge in traditional instruction.

Overall, the findings indicate that incorporating CT into physics teaching is both feasible and educationally valuable, as it opens new avenues for reasoning about motion phenomena through programming. Future work should strengthen the integration between physics and programming, design new ways to assess the learning outcomes that arise from this connection, and develop resources that support its implementation in the classroom.

Keywords:

Computational Thinking, Kinematics, Graph interpretation, Python, PER.

Tabla de contenidos

1	El pensamiento computacional en la enseñanza de la física	1
1.1	Áreas de incorporación de la computación en la enseñanza de la física	2
1.1.1	Modelado y representaciones	2
1.1.2	Prácticas científicas	3
1.1.3	Pensamiento computacional	3
1.2	Inclusión actual del PC en la enseñanza de la física	8
1.2.1	Nivel secundaria	8
1.2.2	Nivel universitario	9
1.3	Objetivos y estructura de este trabajo	10
2	Fundamentos didácticos para la elaboración de las actividades	12
2.1	Marco de prácticas de PC en la enseñanza de la física	13
2.2	Construcción de conocimiento sobre programas	15
2.2.1	Epistemología genética de Jean Piaget	15
2.2.2	Ley General de la Cognición	17
2.2.3	Extensión de la Ley General de la Cognición para el PC	18
2.2.4	Ejemplo de aplicación de la Ley General de la Cognición Extendida	19
2.3	Aprendizaje de cinemática en una dimensión	20
2.3.1	Dificultades conceptuales en cinemática	20
2.3.2	Estrategias para enfrentar dificultades conceptuales	22
3	Diseño de la secuencia didáctica	24
3.1	Primera actividad: Ley horaria en el MRU	24
3.1.1	Diseño de la primera parte (intra→inter)	25
3.1.2	Diseño de la segunda parte (inter→trans)	29

3.2 Segunda actividad: Programa para calcular $x(t)$ y $v(t)$ en el MRUA	33
3.2.1 Diseño de la primera parte (intra→inter)	34
3.2.2 Diseño de la segunda parte (inter→trans)	37
3.3 Tercera actividad: Análisis de movimientos rectilíneos	40
4 Características de los aprendizajes en cinemática	45
4.1 Evaluación mediante TUGK	46
4.1.1 Selección de preguntas del test	46
4.1.2 Aprendizajes en cada categoría	49
4.1.3 Resultados globales del test	64
4.2 Evaluación mediante rúbrica	65
4.3 Discusión de los resultados	70
5 Evaluación del desarrollo del PC	72
5.1 Evaluación sobre algoritmos de MRUA en Python	72
5.1.1 Primera parte: Elaboración de la tabla	73
5.1.2 Segunda parte: Explicación del programa	76
5.1.3 Resultados	79
5.2 Explicación de programa sobre caída libre en Python	82
5.2.1 Primera etapa	82
5.2.2 Evaluación	83
5.2.3 Conclusiones de esta actividad	88
6 Conclusiones y perspectivas	90
Referencias bibliográficas	94
Anexos	101
Anexo 1 Publicaciones y presentaciones en congresos derivadas de la tesis	102

Capítulo 1

El pensamiento computacional en la enseñanza de la física

La comunidad de Physics Education Research (PER) ha mostrado, desde sus inicios, un interés sostenido por comprender cómo se aprende y cómo se enseña la física. Desde los primeros estudios sobre aprendizaje conceptual y cambio de concepciones hace más de medio siglo hasta las actuales investigaciones, este campo ha configurado una comunidad internacional de investigadores que combinan métodos de la educación y de la propia práctica científica (Docktor y Mestre, 2014; McDermott y Redish, 1999; Meltzer y Otero, 2015).

La consolidación de esta comunidad también se refleja en la evolución de sus enfoques y herramientas de investigación y enseñanza. A lo largo de su desarrollo, en PER se han ido incorporando nuevas herramientas y perspectivas en la enseñanza de la física: primero las simulaciones, luego las actividades basadas en datos reales, y más recientemente, la integración de la computación y el PC. En este marco, la incorporación de la computación en la enseñanza de la física se entiende menos como un recurso instrumental y más como una oportunidad para estudiar cómo se extiende y transforma el razonamiento físico cuando los estudiantes utilizan entornos computacionales. Diversos estudios han mostrado que, aunque la computación constituye una práctica habitual en la investigación en física, su integración en la enseñanza de la disciplina avanza con mayor lentitud. En los últimos años, se ha consolidado una línea de investigación dentro de la comunidad de PER orientada a comprender qué significa pensar computacionalmente en física y de qué manera este tipo de razonamiento puede incorporarse a los procesos de enseñanza y aprendizaje (Chonacky y

Winch, 2008; Odden et al. 2019). Desde esta perspectiva, la computación no se concibe únicamente como una herramienta instrumental, sino como una oportunidad para analizar cómo los entornos digitales transforman el razonamiento físico y las formas de representar los fenómenos.

En el ámbito universitario, la integración de la computación en la formación en física ha avanzado principalmente a través de cursos específicos de física computacional, donde se introducen métodos numéricos y de simulación como parte del quehacer profesional. Sin embargo, la incorporación del PC como competencia transversal en el resto de los cursos, en especial en los de nivel introductorio, continúa siendo limitada (Caballero y Merner, 2018). Este escenario plantea el desafío de repensar cómo las prácticas computacionales pueden integrarse en la enseñanza de la física más allá de los cursos especializados de programación o modelado, en un contexto donde la presencia de la computación en la formación universitaria sigue siendo reducida (Caballero y Odden, 2024).

1.1. Áreas de incorporación de la computación en la enseñanza de la física

De acuerdo con la revisión presentada por Odden y Caballero (2023), la literatura reciente en PER sobre la enseñanza con computación puede organizarse en tres grandes tradiciones teóricas: el modelado y las representaciones, las prácticas científicas y el PC. Cada una de ellas se apoya en distintas concepciones epistemológicas de la física y, en consecuencia, propone diferentes justificaciones para la integración de la computación en su enseñanza.

1.1.1. Modelado y representaciones

Esta línea parte de una concepción de la física como un proceso de construcción, refinamiento e interpretación de modelos. En la tradición inaugurada por Hestenes (1987) y desarrollada posteriormente en el paradigma de *Modeling Instruction* (Brewe, 2008), aprender física implica elaborar representaciones abstractas de los fenómenos, analizarlas y coordinarlas entre sí.

Desde esta perspectiva, la computación constituye una herramienta para la producción y coordinación de representaciones, ya que permite generar

gráficos, animaciones, simulaciones y otras formas de visualización que enriquecen el proceso de modelado. Las implementaciones más difundidas, como el currículo *Matter & Interactions* (Chabay y Sherwood, 2008), priorizan la comprensión conceptual a partir de un conjunto limitado de estructuras computacionales (asignación de variables, bucles, condicionales, etc.), buscando que los estudiantes utilicen la programación como medio para explorar y comunicar modelos físicos. Estas experiencias han mostrado que incluso estudiantes sin conocimientos previos de programación pueden construir modelos computacionales significativos cuando la tarea se centra en la representación de los sistemas físicos (Buffler et al. 2008; Caballero et al. 2014).

1.1.2. Prácticas científicas

Una segunda tradición se apoya en la noción de aprendizaje como participación en comunidades de práctica (Lave y Wenger, 1991). Desde esta mirada, aprender física implica apropiarse de las formas de pensar y actuar propias de la comunidad disciplinar, entre ellas el uso de herramientas computacionales.

En esta línea, la computación se incorpora como una práctica científica auténtica: los estudiantes utilizan la programación, el análisis de datos o las simulaciones con propósitos semejantes a los de los físicos profesionales. Esta perspectiva se materializa en propuestas como *Projects and Practices in Physics* (Irving et al. 2017), en las que los estudiantes trabajan en grupos resolviendo problemas abiertos mediante modelado computacional. En este enfoque, la computación no es solo un medio técnico, sino también un espacio para el desarrollo de habilidades sociales, argumentativas y de comunicación científica. De manera similar, otros proyectos, como el *Partnership for Integration of Computation in Undergraduate Physics*, han promovido el intercambio entre docentes e investigadores para diseñar actividades que integren prácticas computacionales auténticas en los cursos introductorios (Caballero et al. 2019).

1.1.3. Pensamiento computacional

La tercera tradición se centra en el PC, que puede entenderse como una tercera epistemología disciplinar dentro de la enseñanza de la física. Esta perspectiva parte de la idea de que la computación amplía las posibilidades de la física para resolver problemas, permitiendo abordar fenómenos de mayor complejidad.

plejidad, variedad y escala. En este sentido, el PC no se limita a incorporar herramientas digitales, sino que representa un modo particular de pensar y producir conocimiento en la disciplina.

Ahora bien, aunque en física el PC se vincula con una forma de razonamiento disciplinar, el concepto proviene originalmente del ámbito de la informática y fue adoptado progresivamente por las disciplinas Science, Technology, Engineering and Mathematics (STEM). Su definición, sin embargo, sigue siendo objeto de debate (Denning y Tedre, 2019; Lodi y Martini, 2021), lo que ha dificultado su aplicación en campos de investigación educativa como PER. En términos generales, y siguiendo la formulación clásica de J. Wing (2010), el PC puede entenderse como los *“procesos mentales implicados en formular problemas y construir soluciones de manera que puedan ser ejecutadas por un agente computacional.”*

Aunque el término PC comenzó a popularizarse a mediados de la década del 2000, algunos autores han rastreado sus raíces en el desarrollo de la ciencia computacional durante los años ochenta. Denning (2017) destaca, por ejemplo, el trabajo de Kenneth Wilson —premio Nobel de Física—, cuyos aportes al modelado computacional ilustran cómo la computación se consolidó como un nuevo modo de producción de conocimiento científico. En esa época, la comunidad científica comenzó a reflexionar sobre los procesos mentales involucrados en la construcción y uso de modelos computacionales, como el diseño de algoritmos, la verificación de resultados y la interpretación de simulaciones. Aunque todavía no se hablaba de pensamiento computacional en los términos actuales, estas prácticas ya implicaban una forma particular de razonar acerca de sistemas complejos mediante herramientas computacionales.

A partir de estas raíces en la ciencia computacional, el interés por el PC se trasladó progresivamente al terreno educativo. Ya en 1980, Papert (1980) analiza cómo el uso de computadoras puede ser una herramienta poderosa para transformar la educación de los niños. Influenciado por la teoría epistemológica de Piaget, establece que mediante el uso de las computadoras los niños construyen conocimiento de una forma más activa, no solamente escuchando la explicación de otro (Papert, 1980). En particular, afirma que las computadoras son herramientas formidables para acceder a conocimiento que antes solo era alcanzable a través del pensamiento formal, y que estos procesos incluyen las habilidades necesarias para convertirse en un pensador formal. Esta perspectiva llevó a interpretar que la computación podía favorecer el desarrollo

de un conjunto de habilidades transferibles a otros ámbitos del conocimiento (Lodi y Martini, 2021).

El concepto se popularizó en el ámbito educativo cuando J. M. Wing (2006), retomando esta línea de pensamiento, lo definió como un conjunto de habilidades fundamentales para cualquier persona. “A la lectura, escritura y aritmética, deberíamos añadir el pensamiento computacional a la capacidad analítica de cada niño” (J. M. Wing, 2006). En su formulación, lo relevante no era tanto precisar una definición exhaustiva, sino destacar el valor del PC como una competencia general aplicable más allá de la informática (Cabezas, 2024). Ejemplos típicos son la descomposición de problemas complejos en partes más simples, la elección de representaciones adecuadas o la búsqueda de soluciones aproximadas (J. M. Wing, 2006).

Desde entonces, múltiples autores han intentado precisar el alcance del concepto (Aho, 2012; Brennan y Resnick, 2012; da Rosa Zipitriá, 2018; Denning y Tedre, 2019; J. M. Wing, 2008), aunque todavía no existe consenso. No obstante, la mayoría coincide en que el PC no se limita a métodos técnicos, sino que involucra procesos mentales y competencias transversales (Lodi et al. 2020). En este sentido, puede considerarse una forma de pensamiento que combina la abstracción, la modelización y la automatización de procesos (Denning y Tedre, 2019; Lodi et al. 2020). Sin embargo, estas formulaciones surgen principalmente del campo de la informática y de la educación general, por lo que su traducción al contexto de la enseñanza de la física requiere precisiones adicionales.

En Uruguay, la competencia en PC forma parte del Marco Curricular Nacional (ANEP, 2022), y Ceibal impulsa un programa específico de pensamiento computacional con su propio Marco Referencial (Plan Ceibal, 2023) y una propuesta concreta para el aula (Plan Ceibal, 2022). Estas iniciativas reflejan la adopción del PC como una competencia transversal en el sistema educativo, en línea con las tendencias internacionales que promueven su integración en la enseñanza de las ciencias y la tecnología.

Aun cuando el PC ha sido ampliamente difundido en la educación científica y tecnológica (Basu et al. 2016; Bocconi et al. 2022; Weintrop et al. 2016), su carácter general ha generado dificultades al momento de aplicarlo a contextos disciplinares específicos (Denning y Tedre, 2019; Lodi y Martini, 2021). En particular, trasladar estas ideas al campo de la enseñanza de la física implica reconocer que la computación no solo introduce nuevas herramientas, sino que

también transforma los modos en que se representa, se razona y se modela dentro de la disciplina (Caballero y Odden, 2024; Skuse, 2019; Winsberg, 2019). Por ello, distintos trabajos en PER han buscado precisar qué significa pensar computacionalmente en física, desarrollando marcos conceptuales y empíricos que articulan las prácticas computacionales con las formas de razonamiento físico (Gambrell, 2024; Odden et al. 2019; Weller et al. 2022).

En este sentido, la comunidad de PER ha comenzado a redefinir el PC en función de las prácticas, representaciones y modos de razonamiento propios de la disciplina. Más que una competencia transversal, se lo entiende como un conjunto de procesos de pensamiento que permiten construir, ejecutar y evaluar modelos computacionales de fenómenos físicos. Esta adaptación disciplinar se ha plasmado en marcos específicos que buscan describir qué significa pensar computacionalmente en física y cómo se manifiesta en la resolución de problemas, el análisis de datos o la comunicación de resultados (Gambrell, 2024; Odden et al. 2019; Weller et al. 2022).

Desde esta perspectiva, el interés por el PC en la enseñanza de la física surge de la necesidad de articular las prácticas computacionales con los objetivos de aprendizaje disciplinar. En PER se han aportado definiciones operativas que lo sitúan no como una habilidad general ni como un conocimiento técnico, sino como una forma particular de razonamiento físico apoyado en representaciones computacionales (Caballero y Odden, 2024; Odden et al. 2019).

En este contexto, Gambrell (2024) propone una definición operativa de PC que permite situarlo dentro de las formas de razonamiento propias de la disciplina. En términos generales, define el pensamiento computacional en física como

una forma especializada de pensamiento que involucra la descomposición de problemas, la identificación o reconocimiento de patrones, la abstracción de sistemas complejos junto con la justificación apropiada, y la construcción de algoritmos orientados a resolver problemas que requieren modelado numérico y resolución mediante el uso de computadoras.

Además, el autor retoma la distinción propuesta por Odden et al. (2019) entre pensamiento computacional y alfabetización computacional. Mientras que el primero alude a los procesos cognitivos implicados en formular y resolver problemas de manera computacional, la alfabetización computacional se refiere

al conocimiento y uso competente de entornos de programación. Siguiendo esta distinción, Gambrell (2024) aclara que no es necesario saber programar para pensar computacionalmente en física, aunque cierta familiaridad con el uso de herramientas computacionales puede facilitar la manipulación de modelos y programas.

En conjunto, esta definición enfatiza que el PC en física trasciende el dominio técnico de la programación y se expresa en la forma en que los estudiantes construyen, ejecutan y documentan cálculos que describen fenómenos físicos de manera eficiente, sistemática y bien justificada. Desde esta perspectiva, el pensamiento computacional constituye un modo particular de razonamiento físico que articula los procesos de modelado numérico con la representación computacional de los sistemas estudiados.

En su trabajo, Gambrell (2024) retoma el marco de prácticas propuesto por Weller et al. (2022) como referencia empírica para caracterizar cómo se expresa el PC en la enseñanza de la física. Estos autores desarrollaron un marco teórico para analizar las prácticas de PC observables en cursos introductorios de física, identificando catorce prácticas agrupadas en seis categorías: extraer conocimiento computacional, construir modelos computacionales, prácticas de datos, depurar, trabajar de forma colaborativa y demostrar disposiciones afectivas hacia la computación. Estas prácticas incluyen acciones como traducir expresiones físicas a código, utilizar lógica condicional, generalizar algoritmos o depurar programas. En conjunto, el marco de Weller et al. (2022) ofrece una descripción empírica de cómo los estudiantes ponen en juego el PC al enfrentarse a tareas de modelado y resolución de problemas.

En el presente trabajo se adopta la definición de Gambrell (2024) para delimitar conceptualmente qué se entiende por PC en física, y se asume que este pensamiento se hace visible a través de las prácticas descritas por Weller et al. (2022). Esta articulación teórica permitirá, en los capítulos siguientes, analizar cómo los estudiantes expresan el PC en situaciones de enseñanza de la cinemática mediadas por el uso de la programación.

1.2. Inclusión actual del PC en la enseñanza de la física

La investigación en torno al PC en la enseñanza de la física ha crecido en los últimos años, aunque su desarrollo es desigual entre los distintos niveles educativos. En esta sección se presenta una síntesis de los principales hallazgos en el ámbito universitario y en la educación secundaria.

1.2.1. Nivel secundaria

En la enseñanza secundaria, la investigación sobre el PC en física es todavía incipiente y se encuentra en una etapa exploratoria (Odden y Caballero, 2023; Orban y Teeling-Smith, 2020; Weller et al. 2022). La mayoría de los trabajos se han centrado en examinar cómo los estudiantes se involucran en prácticas de pensamiento computacional al trabajar con modelos o simulaciones digitales, más que en la enseñanza directa de habilidades de programación (Basu et al. 2016; Weintrop et al. 2016). El marco de prácticas propuesto por Weller et al. (2022) constituye una de las primeras aproximaciones sistemáticas para analizar cómo se manifiesta el PC en cursos de física de este nivel. Basado en observaciones de clase y en la revisión de la literatura, dicho marco evidencia que los estudiantes pueden poner en juego procesos de descomposición, traducción entre representaciones y depuración de algoritmos cuando las tareas ofrecen entornos abiertos y andamiajes adecuados.

Otras experiencias en educación media, inspiradas en el enfoque de Weintrop et al. (2016) y desarrolladas en entornos de aprendizaje por modelado (Basu et al. 2016; Hutchins et al. 2018), han mostrado que el trabajo con simulaciones o programas sencillos puede favorecer tanto la comprensión de conceptos de mecánica como la apropiación de prácticas de indagación científica.

No obstante, la evidencia empírica sigue siendo limitada, y la mayoría de los estudios disponibles corresponden a implementaciones exploratorias de corto plazo (Caballero y Odden, 2024; Odden y Caballero, 2023; Weller et al. 2022). Hasta el momento no se han desarrollado investigaciones longitudinales que permitan evaluar el impacto sostenido del PC en los aprendizajes de física en secundaria (Basu et al. 2016; Orban y Teeling-Smith, 2020; Weintrop et al. 2016). Diversas revisiones coinciden en señalar la necesidad de avanzar hacia estudios sistemáticos que analicen cómo se desarrolla el PC en distintos niveles

educativos y contextos disciplinares (Bocconi et al. 2022; Odden y Caballero, 2023). Esta escasez de investigaciones evidencia la importancia de continuar explorando el PC en la enseñanza de la física en educación media, particularmente en situaciones donde la programación se integra como herramienta para construir y razonar sobre modelos físicos, como se propone en este trabajo.

1.2.2. Nivel universitario

En el nivel universitario, las investigaciones sobre el PC en física se han desarrollado principalmente en el marco de propuestas que incorporan el modelado computacional y la programación como medios para favorecer el razonamiento físico. Si bien muchos de estos trabajos no utilizan explícitamente el término PC, sus resultados permiten inferir cómo los estudiantes movilizan procesos característicos de este tipo de pensamiento al enfrentarse a tareas de modelado numérico. Diversos estudios muestran que incluso los estudiantes sin experiencia previa en programación pueden aprender a utilizar el modelado computacional como parte de su aprendizaje de la física (Buffler et al. 2008; Caballero et al. 2012), y que el trabajo con modelos programados favorece la comprensión conceptual y la capacidad de conectar representaciones (Aiken et al. 2012; Svensson et al. 2020).

Los entornos de aprendizaje más investigados, como *Matter & Interactions*, *Projects and Practices in Physics* o las experiencias desarrolladas en el marco de la *Partnership for Integration of Computation in Undergraduate Physics*, ofrecen contextos donde los estudiantes construyen y analizan modelos computacionales que representan sistemas físicos (Hutchins et al. 2020; Odden et al. 2019). En estos entornos se promueven formas de razonamiento que articulan la física, la matemática y la computación, y que pueden interpretarse como manifestaciones del PC en la disciplina.

A pesar de estos avances, la literatura también destaca importantes limitaciones. Caballero y Merner (2018) muestran que, si bien una parte de los docentes universitarios incorpora actividades computacionales, la adopción formal de la computación en los programas sigue siendo reducida. Entre los obstáculos más señalados se encuentran la falta de tiempo curricular, la escasa formación docente y la ausencia de instrumentos de evaluación validados. En este sentido, la tesis de Gambrell (2024) constituye uno de los primeros esfuerzos explícitos por definir y evaluar el PC en física introductory, desarro-

llando un instrumento que busca medir la forma en que los estudiantes razonan computacionalmente al resolver problemas físicos.

La revisión de la literatura muestra que, si bien existen avances significativos en la integración del PC en la enseñanza universitaria de la física, la evidencia empírica en el nivel secundario sigue siendo escasa y fragmentaria (Odden y Caballero, 2023; Weller et al. 2022). En particular, aún no se cuenta con estudios sistemáticos que analicen cómo los estudiantes de este nivel desarrollan prácticas de pensamiento computacional al abordar contenidos de mecánica mediante programación. Este trabajo se propone contribuir a ese campo emergente de investigación, centrando el análisis en el desarrollo del PC en actividades de enseñanza de la cinemática.

1.3. Objetivos y estructura de este trabajo

El objetivo general de este trabajo es explorar la integración del PC en la enseñanza de la física, particularmente en actividades que vinculan la programación con el estudio de la cinemática.

A partir de este objetivo general, la pregunta de investigación que orienta este estudio es: ¿Es posible desarrollar el pensamiento computacional en actividades que vinculan cinemática con programación?

De esta pregunta se derivan las siguientes preguntas específicas:

- ¿Cómo puede diseñarse una secuencia de actividades que integre el PC con el estudio de la cinemática en educación secundaria?
- ¿Qué características presentan los aprendizajes de los estudiantes en cinemática durante la implementación de la secuencia propuesta?
- ¿Qué prácticas de pensamiento computacional se manifiestan durante la resolución de las actividades y de qué manera se expresan en las producciones de los estudiantes?

El trabajo se organiza en seis capítulos. El capítulo 2 desarrolla los fundamentos teóricos que orientan el diseño de la secuencia, articulando el marco de prácticas de PC con la epistemología genética de Piaget (1968) y con la extensión de la Ley General de la Cognición propuesta por da Rosa Zipitriá (2018). El capítulo 3 describe la secuencia de actividades diseñada para integrar el PC en el estudio de la cinemática. En el capítulo 4 se analiza la comprensión de los estudiantes en cinemática a partir de la evaluación realizada antes y después de

la implementación de la secuencia. El capítulo 5 examina las prácticas de pensamiento computacional que emergen durante el trabajo con las actividades. Finalmente, el capítulo 6 presenta las conclusiones generales y las proyecciones a futuro del estudio.

Capítulo 2

Fundamentos didácticos para la elaboración de las actividades

En el capítulo anterior adoptamos la definición de Gambrell (2024) para delimitar conceptualmente el PC en física aceptando que dicho pensamiento se manifiesta empíricamente mediante las prácticas descritas por Weller et al. (2022). Sobre esa base, este capítulo operacionaliza ese marco de prácticas para orientar el diseño de las actividades, articulándolo con la epistemología genética de Piaget (1968) y con la extensión de la Ley General de la Cognición propuesta por da Rosa Zipitría (2018).

Dado que las actividades de este trabajo se centran en la cinemática, consideramos el marco de prácticas de Weller et al. (2022) no solo como una referencia para promover el desarrollo del PC, sino también como una herramienta para analizar cómo los estudiantes construyen y refinan modelos computacionales de movimiento. En este sentido, la programación se integra al estudio de la cinemática como un medio para representar, explorar y explicar fenómenos físicos, permitiendo observar de qué manera las prácticas de PC contribuyen a la comprensión de los conceptos de posición, velocidad y aceleración.

El diseño de las actividades, por tanto, se apoya en marcos teóricos que permiten comprender cómo los estudiantes construyen conocimiento al interactuar simultáneamente con fenómenos físicos y con representaciones computacionales. Estos marcos proporcionan las bases para analizar de qué modo las acciones de los estudiantes, como descomponer un problema, construir algoritmos o depurar un modelo, se vinculan con la comprensión de los conceptos de cinemática. En lo que sigue, presentamos los fundamentos didácticos

que orientaron la elaboración de la secuencia: primero, el marco de prácticas de pensamiento computacional propuesto por Weller et al. (2022); luego, los aportes de la epistemología genética de Piaget (1968) y la Ley General de la Cognición extendida (da Rosa Zipitriá, 2018); y, finalmente, los principales hallazgos sobre el aprendizaje de la cinemática en una dimensión y sus dificultades conceptuales.

2.1. Marco de prácticas de PC en la enseñanza de la física

El marco de Weller et al. (2022) fue construido a partir de una revisión de la literatura y del análisis de videos de clases de física de nivel secundario en las que se integra computación (VPython), además de sucesivas rondas de revisión con especialistas en enseñanza de la computación y en PER. El resultado es un conjunto de 14 prácticas de PC agrupadas en 6 categorías (Tabla 2.1) que utilizamos como lente analítica y como guía de diseño de las actividades de este trabajo.

La categoría **extraer conocimiento computacional** se refiere a la capacidad de identificar los componentes esenciales de un modelo computacional, visualizándolo como una abstracción de un fenómeno físico real. Dentro de esta categoría, una de las prácticas es descomponer, que consiste en dividir un problema computacional en tareas manejables. La otra práctica es destacar y poner en plano, que implica identificar los aspectos más relevantes de una tarea computacional para favorecer la comprensión, enfocarse en los elementos esenciales del código y reconocer comportamientos inesperados.

Construir modelos computacionales implica utilizar la computadora para desarrollar una representación abstracta de un sistema o fenómeno físico. Entre sus cinco prácticas se encuentra traducir física a código, que supone adaptar un modelo analítico a un entorno computacional. La construcción de algoritmos implica diseñar una secuencia de pasos ordenados para modelar un fenómeno físico. Por su parte, aplicar la lógica condicional refiere a planificar una secuencia lógica de eventos y editar sentencias condicionales en un problema. Utilizar la generalización consiste en reaprovechar enfoques, algoritmos o códigos previos para aplicarlos a un nuevo modelo. Finalmente, agregar complejidad al modelo implica enriquecer progresivamente un modelo compu-

Tabla 2.1: Prácticas de pensamiento computacional según el marco de Weller et al. (2022). Las últimas tres prácticas están dentro de una categoría exclusiva para cada una.

Categoría	Prácticas de Pensamiento Computacional
Extraer conocimiento computacional	Descomponer Destacar y poner en primer plano
Construir modelos computacionales	Traducir física a código Construcción de algoritmos Aplicar la lógica condicional Utilizar la generalización Aregar complejidad al modelo
Prácticas de datos	Elegir la forma de representar datos Generar datos Manipular datos Analizar datos
Depurar	Identificar y resolver errores o comportamientos inesperados en el código.
Trabajar en grupos en modelos computacionales	Colaborar activamente en la resolución de tareas computacionales.
Demostrar disposiciones afectivas hacia la computación	Expresar, reconocer y manejar emociones como frustración, perseverancia o entusiasmo ante desafíos computacionales.

tacional incorporando nuevas características físicas que lo hagan más completo, realista o sofisticado.

La categoría **prácticas de datos** reúne diversas formas en que los estudiantes pueden obtener y trabajar con información mediante la computación. Una de sus prácticas es elegir la forma de representar datos, lo que implica seleccionar la mejor técnica, enfoque o herramienta para comunicar los resultados. Generar datos consiste en producir información a través de la ejecución de un modelo computacional. En tanto, manipular datos refiere al procesamiento, organización y selección de un conjunto de datos con el fin de prepararlos para su análisis. Por último, analizar datos implica interpretar y extraer significado a partir de un conjunto de datos o de los resultados obtenidos por medio de un modelo computacional.

Las tres categorías restantes tienen una práctica cada una. **Depurar** implica enfrentar comportamientos inesperados o errores en el código y encontrar soluciones para corregirlos. No se limita a errores fatales que impiden la ejecución del programa, sino que también abarca ajustes cuando el modelo no se comporta como se espera según la física. **Trabajar en grupo sobre modelos computacionales** refiere a colaborar activamente con otros para construir o mejorar un modelo, intercambiando ideas, resolviendo problemas en conjunto y compartiendo decisiones sobre el código. Finalmente, **demostrar disposiciones afectivas hacia la computación** implica gestionar las emociones que surgen durante la actividad computacional, como la frustración o el entusiasmo, y perseverar a pesar de los obstáculos.

En este trabajo, las categorías y prácticas del marco de Weller et al. (2022) cumplen un doble papel: orientan decisiones de diseño y proporcionan criterios de análisis para identificar cómo los estudiantes ponen en juego el PC durante la secuencia.

2.2. Construcción de conocimiento sobre programas

El marco de prácticas de Weller et al. (2022) describe lo que los estudiantes hacen cuando ponen en juego el PC en la resolución de tareas de modelado computacional. Para comprender cómo se construye ese conocimiento durante la interacción con la computadora es necesario incorporar una perspectiva que dé cuenta de los procesos cognitivos que subyacen a dichas prácticas. En este sentido, la epistemología genética de Piaget (1968) y la extensión de la Ley General de la Cognición propuesta por da Rosa Zipitriá (2018) ofrecen un marco para analizar cómo los estudiantes elaboran y comprenden programas y algoritmos a medida que actúan, reflexionan y generalizan sobre sus propias acciones en el entorno computacional.

2.2.1. Epistemología genética de Jean Piaget

La Epistemología genética es una teoría elaborada por Jean Piaget, que apunta a explicar la construcción del conocimiento en amplios ámbitos y niveles de desarrollo. Sus puntos centrales son el estudio de la construcción del conocimiento como proceso y cómo es la transición de un nivel de conocimiento

a otro más alto. Su hipótesis central es la existencia de un paralelismo entre el progreso realizado en la organización lógica y racional del conocimiento y los correspondientes procesos psicológicos formativos. (Piaget, 1968)

La teoría es fundamentada a partir de información que proviene de dos fuentes. Por un lado, mediante estudios empíricos (Piaget, 1975) sobre la construcción de conocimiento de niños y adolescentes. En segundo lugar, mediante el análisis histórico crítico de la historia de la ciencia en el desarrollo de los conceptos y teorías, desarrollado por Piaget y García (1989).

Para describir las transiciones en las que se construye el conocimiento Piaget y García utilizan una tríada de etapas, llamadas *intra*, *inter* y *trans*. En la etapa *intra* el sujeto se centra en objetos y elementos aislados, en la *inter* tiene en cuenta las relaciones entre los objetos y sus transformaciones, y en la *trans* se construye sobre estructuras generales, que integran a elementos generalizados y sus transformaciones, e incluye a las construcciones de las etapas anteriores como casos aislados. (da Rosa y Gómez, 2020)

Según Piaget (1968),

el conocimiento humano es esencialmente activo. Conocer es asimilar la realidad en sistemas de transformaciones. Conocer es transformar la realidad para comprender cómo se produce un determinado estado. En virtud de este punto de vista, me opongo a la idea del conocimiento como una copia, una copia pasiva, de la realidad. De hecho, esta noción se basa en un círculo vicioso: para hacer una copia, debemos conocer el modelo que estamos copiando, pero según esta teoría del conocimiento, la única manera de conocer el modelo es copiándolo, lo que nos atrapa en un círculo sin salida, incapaces de saber si nuestra copia del modelo se asemeja o no al modelo original. Desde mi punto de vista, conocer un objeto no significa copiarlo, sino actuar sobre él. Significa construir sistemas de transformaciones que puedan aplicarse a este objeto o con él. Conocer la realidad significa construir sistemas de transformaciones que correspondan, de manera más o menos adecuada, a la realidad. Estos sistemas son más o menos isomorfos a las transformaciones de la realidad. Las estructuras de transformación que constituyen el conocimiento no son copias de las transformaciones en la realidad; son simplemente modelos isomorfos posibles, entre los cuales la experiencia nos permite elegir. El conocimiento, entonces, es un sistema

de transformaciones que se vuelve progresivamente más adecuado.

El problema más general de todo el desarrollo epistémico radica entonces en determinar el rol de la experiencia y las estructuras operacionales del individuo en la construcción del conocimiento, y en examinar los instrumentos mediante los cuales se ha adquirido antes de su formalización. Este problema fue estudiado por Piaget en sus experimentos sobre psicología genética, y dieron lugar a la ley general de la cognición. (Piaget, 1964)

2.2.2. Ley General de la Cognición

La ley general de la cognición es representada por Piaget mediante el diagrama

$$C \leftarrow P \rightarrow C' \quad (2.1)$$

donde P representa a la periferia, que es la reacción más exterior e inmediata que tiene el sujeto al enfrentarse a los objetos para resolver un problema. Esta acción está asociada a seguir un objetivo y conseguir resultados sin ser consciente de las acciones realizadas ni las razones por las que fueron exitosas. Las flechas representan los mecanismos de pensamiento, mediante los que el sujeto se hace consciente de sus acciones (C en el diagrama), y los cambios que ellas producen en los objetos y sus propiedades intrínsecas (C' en el diagrama). (Piaget, 1964)

Piaget también describe los elementos cognitivos involucrados en estos procesos, llamados abstracción reflexiva y generalización constructiva. La abstracción reflexiva es un proceso en el que se proyecta al plano del pensamiento las relaciones establecidas en el plano de acciones, y luego se reconstruyen las relaciones en el plano de pensamiento añadiendo la comprensión de las condiciones y motivaciones. El motor de este proceso es denominado por Piaget como la búsqueda de razones de éxito (o fracaso). (Piaget, 1964)

Tras comprender un método utilizado para resolver un problema, el razonamiento de los estudiantes intenta generalizar lo que ha construido exitosamente a otras situaciones, mediante una generalización inductiva. Mediante la generalización constructiva, el individuo comprende las nuevas condiciones que dan lugar a estructuras, lo que permite estudiar nuevos elementos y considerar las etapas previas como casos particulares (Piaget, 1964). da Rosa aplica la ley general de Piaget para investigar la construcción de conocimiento sobre estructuras de datos y algoritmos (da Rosa, 2015) y propone una extensión de

la ley para abarcar la construcción de conocimiento sobre programas (da Rosa y Aguirre, 2018; da Rosa y Gómez, 2020; Gómez y da Rosa, 2022), que se describe en la siguiente sección.

2.2.3. Extensión de la Ley General de la Cognición para el PC

La ley general de la cognición extendida fue formulada por da Rosa y Aguirre (2018) para explicar la construcción de conocimiento sobre algoritmos y estructuras de datos. La extensión es necesaria ya que la ley general no es suficiente para describir los casos en que una persona debe interactuar con una computadora. (da Rosa Zipitria, 2018)

La misma se inspira en Papert (1980), que establece que “programar un autómata que resuelve un problema comienza haciendo que el estudiante reflexione sobre cómo realiza por sí mismo lo que le gustaría que el autómata hiciera”. Esta afirmación se puede resumir estableciendo que programar un autómata comienza haciendo que uno reflexione sobre

cómo uno lo hace por sí mismo
lo que le gustaría que hiciera el autómata

Es decir, que “para programar un autómata los estudiantes deben establecer una relación causal entre el algoritmo (ellos actuando sobre los objetos) y la ejecución del programa (la computadora actuando sobre los estados)” (da Rosa y Aguirre, 2018; Gómez y da Rosa, 2022).

La ley general de la cognición extendida se describe con el diagrama

$$\begin{array}{c} C \leftarrow P \rightarrow C' \\ \hline newC \leftarrow newP \rightarrow newC' \end{array}$$

donde $newP$ representa una periferia centrada en las acciones del sujeto y los objetos sobre los que actúa. El centro $newC$ representa la conciencia sobre la ejecución de las instrucciones del programa y $newC'$ sobre las modificaciones que producen en la representación de las estructuras de datos. La construcción de conocimiento sobre los métodos (algoritmos) y los objetos (estructuras de datos) ocurre en la interacción entre P , C y C' . De forma análoga, la construcción de conocimiento sobre la ejecución de un programa subyace en los me-

canismos internos de pensamiento, representados por las flechas entre $newP$, $newC$ y $newC'$. (da Rosa y Aguirre, 2018; da Rosa Zipitría, 2018; Gómez y da Rosa, 2022)

La primera fila del diagrama explica entonces la construcción de conocimiento sobre algoritmos, mientras que la segunda explica la construcción de conocimiento sobre programas. Además, el vínculo entre ellos es indicado mediante la llave que se encuentra entre ellos. (da Rosa Zipitría, 2018; Gómez y da Rosa, 2022)

2.2.4. Ejemplo de aplicación de la Ley General de la Cognición Extendida

En la sección 3.1.2 se explica la aplicación de la ley extendida para el problema trabajado en clase, pero para finalizar esta descripción teórica, incluimos un ejemplo puntual sobre cómo se puede utilizar esta ley para desarrollar una actividad en la que un estudiante construya conocimiento sobre un programa sencillo. Analizaremos el caso de la elaboración de un programa en el que se ordena un *array* desordenado, y quiénes son en este caso P , C , C' , $newP$, $newC$ y $newC'$.

Según esta ley, para que un estudiante elabore el programa debería en primer lugar reflexionar sobre cómo hace por sí mismo esta tarea en una situación análoga. Por lo tanto, en una actividad se les puede proponer que ordenen de menor a mayor un mazo con cartas, para que a partir de la reflexión sobre cómo ordenan las cartas piensen cómo elaborar un programa que haga lo mismo con un *array*.

En este caso, un estudiante se encuentra en la periferia P cuando logra ordenar el mazo de cartas, pero no es consciente sobre las acciones que realizó ni de los cambios que producen sobre los objetos.

Supongamos que para ordenar el mazo de cartas el estudiante busca la carta más pequeña de todo el mazo, la quita de él y la apoya sobre un nuevo mazo (que al principio no tiene ninguna carta), y que repite estos pasos mientras queden cartas. Para esta forma de resolver la tarea, la flecha del diagrama que apunta hacia el centro C representa los mecanismos mediante los que el estudiante se hace consciente de la coordinación de acciones que realizó para ordenar el mazo de cartas, y la que va hacia C' los cambios que cada una de esas acciones produce sobre los objetos (cada carta, el mazo de cartas que tenía

y el nuevo).

En una segunda instancia, los estudiantes deberían realizar un programa que ordene un *array* desordenado, siguiendo un algoritmo análogo al descrito para ordenar las cartas. En esta instancia, la flecha que va de *newP* a *newC* representa la toma de conciencia sobre la coordinación de acciones ejecutadas por la computadora (algoritmo), y la que va hacia *newC'* los cambios que estas acciones producen sobre las estructuras de datos (que dependen del lenguaje utilizado, pero al menos uno de esos datos debe ser el *array* que fue representado mediante el mazo).

2.3. Aprendizaje de cinemática en una dimensión

El estudio del movimiento y sus causas es fundamental en nuestra interacción con el mundo. Nos permite comprender fenómenos relacionados al funcionamiento de vehículos, la trayectoria de una pelota al ser lanzada en un deporte, a las formas en las que nos movemos las personas e incluso al funcionamiento del sistema solar entre muchos otros ejemplos. Esta importancia se refleja en los contenidos de la enseñanza a todo nivel. En particular, en la enseñanza media los conceptos básicos de cinemática son un contenido clave dentro de los cursos de física de ciclo básico, bachillerato y enseñanza técnica. A su vez, son abordados nuevamente en los cursos universitarios de física general, involucrando a todas las carreras relacionadas a las ciencias naturales e ingeniería. Sin embargo, incluso luego que los estudiantes transitan varios cursos en los que se trabajan estos temas, se ha reportado que tienen importantes dificultades para comprender los conceptos fundamentales de cinemática. (Beichner, 1994; Bollen et al. 2016; Ceuppens et al. 2019; McDermott et al. 1987; Trowbridge y McDermott, 1980, 1981)

2.3.1. Dificultades conceptuales en cinemática

Ya desde 1980 se han reportado dificultades que tienen los estudiantes para comprender los conceptos básicos de cinemática en una dimensión. Trowbridge y McDermott (1980) hicieron una de las investigaciones pioneras, centrada en el entendimiento de la velocidad. Encontraron que muchos estudiantes la confunden con la posición, ya que en algunos ejemplos pensaban que en los

instantes en que dos objetos tienen la misma posición también tienen la misma velocidad. Además encontraron una dificultad en comprender que la velocidad se puede definir para un instante, y no solo en un intervalo. En un segundo artículo (Trowbridge y McDermott, 1981) analizaron también las dificultades en la comprensión de la aceleración, y encontraron que muchos estudiantes la confunden con la posición y con la velocidad, o si bien reconocen que está vinculada a la variación de la velocidad ignoran el intervalo de tiempo en el que sucede.

Pocos años después se estudiaron las dificultades de los estudiantes al conectar conceptos físicos con representaciones gráficas. McDermott et al. (1987) reportaron que en gráficas de posición, velocidad y aceleración en función del tiempo a los estudiantes les cuesta distinguir entre la pendiente y la ordenada, interpretar cambios en la pendiente y cambios en la ordenada, vincular gráficas de distintas magnitudes para un mismo movimiento, asociar descripciones de movimientos con elementos de las gráficas e interpretar el área debajo de la gráfica. En general, los errores que cometen los estudiantes al estudiar estas gráficas están vinculados a pensar que la gráfica es una imagen literal de la situación y a confundir el significado de la pendiente con el de la ordenada (Beichner, 1994).

En esta línea, Beichner (1994) elaboró un test múltiple opción para evaluar el entendimiento de las gráficas en cinemática, el Test of Understanding Graphs in Kinematics (TUGK). Al implementarlo se volvieron a identificar los mismos errores, pero también se encontraron elementos nuevos de algunos de ellos. Por ejemplo, respecto a las dificultades en el cálculo de pendientes, encontró que a los estudiantes les cuesta mucho más cuando las rectas no pasan por el origen. Este aspecto está muy vinculado a calcular la pendiente dividiendo el valor de la ordenada con el de la abscisa en lugar de dividiendo las variaciones de ellas (por ejemplo, haciendo $v = x/t$ en vez de $v = \Delta x/\Delta t$).

Investigaciones más recientes han mostrado que muchas de las dificultades clásicas en la comprensión de la cinemática persisten incluso en los niveles medio y universitario. Bollen et al. (2016) analizaron las interpretaciones de estudiantes universitarios sobre gráficos lineales de posición en función del tiempo y hallaron que, aunque algunos errores como la confusión entre ordenada y pendiente son menos frecuentes, siguen siendo muy comunes las confusiones de intervalo o punto para determinar la pendiente. A su vez siguen siendo comunes las interpretaciones pictóricas, donde el gráfico es entendido como una imagen

literal del movimiento. Estos resultados evidencian una débil conexión entre la comprensión matemática de las funciones lineales y su aplicación en contextos físicos. En la misma línea, Ceuppens et al. (2019) estudiaron a estudiantes de noveno grado al resolver problemas de funciones lineales en matemáticas y en cinemática unidimensional. El estudio mostró que los estudiantes tienden a tratar la física y la matemática como dominios conceptualmente separados, con escasa integración entre ambos, lo que limita la posibilidad de aplicar el conocimiento matemático a la interpretación de fenómenos cinemáticos.

En conjunto, los estudios coinciden en que la interpretación de gráficos sigue representando un desafío central en el aprendizaje de la cinemática. Los errores más frecuentes revelan dificultades para vincular la forma del gráfico con las características del movimiento, lo que limita una comprensión conceptual del fenómeno.

2.3.2. Estrategias para enfrentar dificultades conceptuales

Diversas investigaciones han mostrado que las dificultades en la comprensión de la cinemática pueden abordarse eficazmente mediante metodologías de enseñanza que promuevan la participación activa de los estudiantes. Las estrategias de aprendizaje activo, como la discusión entre pares o la resolución colaborativa de problemas, producen mejoras significativas en la comprensión conceptual de la física y en el desarrollo del razonamiento científico (Crouch y Mazur, 2001; Fraser et al. 2014; Hake, 1998).

También se ha evidenciado que la visualización en tiempo real de los movimientos favorece la interpretación de las gráficas cinemáticas. Los estudiantes que observan o manipulan representaciones gráficas en tiempo real muestran una comprensión más profunda de las relaciones entre posición, velocidad y tiempo (Duijzer et al. 2019). En esta misma línea, la incorporación de laboratorios activos en los que los estudiantes registran y analizan movimientos reales mediante sensores ha demostrado mejorar sustancialmente la comprensión de estos conceptos (Sokoloff et al. 2007).

Otra línea de trabajo destaca el potencial del modelado computacional como medio para promover la comprensión de los fenómenos cinemáticos, al permitir que los estudiantes exploren de manera interactiva cómo las variaciones en los parámetros afectan las representaciones gráficas del movimiento

(Aiken et al. 2013; Caballero et al. 2014).

Por último, el uso de entornos lúdicos y videojuegos educativos se ha mostrado eficaz para aumentar la motivación y el compromiso de los estudiantes, sin comprometer la profundidad conceptual del aprendizaje (Panthalookaran, 2018; Rose et al. 2016). Este tipo de propuestas puede favorecer la implicación sostenida de los estudiantes y facilitar la construcción activa de significados sobre el movimiento.

Capítulo 3

Diseño de la secuencia didáctica

Este capítulo presenta el proceso de diseño y la secuencia de actividades elaborada para integrar el PC en la enseñanza de la cinemática en educación secundaria. El diseño se fundamenta en los marcos teóricos desarrollados en los capítulos anteriores y busca mostrar cómo es posible articular las prácticas de pensamiento computacional con el estudio del movimiento en una dimensión.

Las experiencias se realizaron con estudiantes de nivel K9 (equivalente al noveno grado en Uruguay), de una institución privada de la zona metropolitana del país con estudiantes de perfil socioeconómico mayoritariamente medio-alto. Cada grupo de clase estuvo conformado por 32 estudiantes, organizados en equipos de cuatro integrantes. Las actividades se centraron en el análisis y modelado de situaciones de movimiento rectilíneo.

3.1. Primera actividad: Ley horaria en el MRU

Esta primera actividad está dirigida a estudiantes que han trabajado previamente con los conceptos básicos de cinemática en una dimensión, como posición, desplazamiento, velocidad y aceleración, y con nociones elementales de programación en Python, como variables, tipos de datos, listas y bucle while. El objetivo es que los estudiantes elaboren un programa que construya una lista de tiempos, calcule las posiciones correspondientes en un Movimiento Rectilíneo Uniforme (MRU) y represente gráficamente la posición en función del tiempo.

En una primera etapa, los estudiantes reciben la posición inicial y la veloci-

dad de una persona que se mueve con velocidad constante, con la consigna de completar una tabla que relacione los valores de tiempo y posición correspondientes a este movimiento. Luego deben realizar una explicación en lenguaje natural sobre cómo realizaron la tabla. La etapa está centrada en el proceso de construcción de conocimiento conceptual a partir de uno instrumental que corresponde a la transición de la etapa intra a la inter. Se espera que mediante la búsqueda de razones del éxito al realizar la actividad los estudiantes logren conceptualizar el algoritmo utilizado y que su pensamiento transite de la periferia P (reacción más inmediata al enfrentarse al problema) a los centros C (reconocer la coordinación de acciones realizadas) y C' (los cambios que esas acciones producen en los objetos).

La segunda etapa está pensada para la construcción de conocimiento formal a partir del conceptual que corresponde a una transición de la etapa inter a la trans. Aquí se buscará que el estudiante obtenga una solución general utilizando un lenguaje formal, y que la pueda ejecutar una computadora en lugar de él mismo. Según la ley general de la cognición extendida en este proceso se espera que el pensamiento del estudiante transite de $newP$ (el conocimiento conceptual del algoritmo) hacia los centros $newC$ (reconocer como la computadora ejecuta las instrucciones del programa) y $newC'$ (reconocer los cambios realizados sobre las estructuras de datos).

3.1.1. Diseño de la primera parte (intra→inter)

En una primera instancia, se les otorga a los estudiantes la siguiente consigna:

Una persona se encuentra caminando en línea recta y con velocidad constante. Su posición inicial es $x_i = 8$ m y su velocidad $v = 3$ m/s. Realiza una tabla en la que se indique la posición de la persona para cada valor de tiempo, donde tomes valores de tiempo cada 0,5 s entre los valores 0 s y 5 s.

Se espera que los estudiantes logren realizar esta actividad sin mayores dificultades dado que ya han resuelto problemas de cinemática en una dimensión. Al trabajar en grupos de estudiantes es posible que algunos no puedan llevarla adelante. En ese caso, el docente debería recordarles qué quiere decir que la velocidad de una persona sea 3 m/s, y cómo puede utilizar este dato para elaborar la tabla. Al finalizar la actividad, deberían obtener la Tabla 3.1.

Tabla 3.1: Posición en función del tiempo de una persona que se mueve con velocidad constante $v = 3 \text{ m/s}$ y posición inicial $x_i = 8 \text{ m}$.

t(s)	x(m)
0	8
0,5	9,5
1	11
1,5	12,5
2	14
2,5	15,5
3	17
3,5	18,5
4	20
4,5	21,5
5	23

Es frecuente que algunos grupos comiencen la lista de posiciones con 0 en lugar de 8 m. Este comportamiento parece estar asociado a la familiaridad con ejemplos previos en los que la posición inicial coincide con el origen, por lo que los estudiantes tienden a asumir que las tablas de valores deben iniciarse siempre en cero. En estos casos, suele bastar con que el docente les pregunte si la persona realmente se encontraba en la posición 0 al comenzar el movimiento o si su posición inicial era diferente, para que reconozcan la necesidad de ajustar la tabla.

Consideraremos como periferia (P) los casos en que los estudiantes logran completar la tabla, ya sea en el primer intento o con ayuda de las preguntas orientadoras. Esto se debe a que el interés de la actividad está en la conceptualización del algoritmo una vez que los estudiantes ya saben resolver el problema, y no en el proceso mediante el cual logran hacerlo por primera vez.

En el problema intervienen siete magnitudes físicas. Cinco de ellas, la posición inicial (x_i), la velocidad (v), el tiempo inicial (t_i), el tiempo final (t_f) y el paso de tiempo (dt), se mantienen constantes durante la resolución. En cambio, las otras dos, que corresponden a las secuencias de tiempos y de posiciones, varían a lo largo del proceso y se registran en la tabla.

Las acciones necesarias para resolver este problema son:

- Agregar en ambas secuencias los valores iniciales indicados.
- Agregar un valor en la secuencia de tiempos, que sea igual al anterior + dt .

- Agregar en la secuencia de posiciones el valor anterior + el desplazamiento en dt (1,5 m en este caso).
- Repetir los dos pasos anteriores hasta que el último valor de la secuencia de tiempos sea mayor o igual al valor final.

Otro camino muy similar para resolver el problema es:

- Agregar en ambas secuencias los valores iniciales indicados.
- Agregar un valor en la secuencia de tiempos, que sea igual al anterior + dt .
- Repetir el paso anterior hasta que el último valor de la secuencia de tiempos sea mayor o igual al valor final.
- Agregar en la secuencia de posiciones el valor anterior + el desplazamiento en dt (1,5 m en este caso).
- Repetir el paso anterior hasta que la secuencia de posiciones tenga la misma cantidad de elementos que la de los tiempos.

Llamaremos Algoritmo 1 al primero y Algoritmo 2 al segundo, y analizaremos posibles respuestas de los estudiantes para cada uno de ellos.

En cada una de estas acciones se agrega al menos un valor al final de una o las dos secuencias. En consecuencia, el último valor de esa secuencia cambia, y este aspecto es clave ya que ese valor se utilizará posteriormente para calcular el siguiente. Este valor también es importante porque es el que se debe utilizar para comparar con el valor final, y de esta forma decidir si terminar con la repetición de acciones. Al agregar valores en una secuencia también se altera su cantidad de elementos, y este aspecto es importante cuando se quiere resolver el problema mediante el Algoritmo 2 ya que en el último paso se repiten acciones hasta que la cantidad de elementos de la secuencia de posiciones sea igual que en la de tiempos.

Tras completar la tarea, los estudiantes deben explicar el procedimiento seguido. A continuación se presenta una descripción representativa considerada correcta:

En primer lugar debemos poner los primeros valores en la tabla, que son 0 en la de tiempo y 8 en la de posición porque es la posición inicial. Luego debemos agregar un valor en la columna de tiempos, que sea el anterior + dt y en la columna de posiciones debemos agregar el valor de la posición anterior + 1,5 m porque es lo que

se desplaza la persona en dt . Estos dos pasos los debemos repetir hasta llegar a 5 s en la columna de los tiempos.

Esta descripción correspondería a un grupo que resuelva el problema mediante el Algoritmo 1. Es muy probable que en la primera descripción tengan algunos errores, por lo que el docente debería orientarlos con algunas preguntas hasta llegar a una descripción correcta.

Por ejemplo, es probable que algunos estudiantes no se den cuenta de que deben agregar los valores iniciales en la tabla antes de comenzar con las repeticiones. En ese caso el algoritmo fallaría ya que no se pueden agregar valores realizando una operación con el valor anterior de la tabla si originalmente la tabla está vacía. En este caso, habría que preguntarles si se puede realizar la operación tiempo anterior + dt o posición anterior + 1,5 m si no existe un tiempo/posición anterior la primera vez que se intenta realizarlo.

También podría pasar que en una de las descripciones se diga que los pasos se deben repetir 10 veces en vez de hasta que el último valor de tiempo sea 5 s. Si bien esto es correcto, solo funciona en casos muy particulares. Además, cuando una persona realiza la actividad probablemente no decida repetir lo realizado 10 veces, sino que realiza las repeticiones hasta llegar al valor final de tiempo. Es decir, es muy probable que un estudiante describa un algoritmo correcto para realizar esta actividad pero que no coincida con el que utilizó. En este caso habría que preguntarle si cuando realizó la actividad realmente hizo ese cálculo o si decidió dejar de repetir acciones con otro criterio.

Es posible que algunos estudiantes modifiquen los valores de tiempo y posición en lugar de agregarlos a la tabla. Ante esta situación, el docente puede intervenir preguntando cuántos valores contendría la tabla si se reemplazara continuamente el último registro, de modo que los estudiantes reconozcan la necesidad de conservar todos los datos generados.

A su vez algún estudiante puede establecer que se realicen repeticiones hasta que el último valor de la secuencia de tiempos sea mayor al final en vez de mayor o igual. En este caso le preguntaría si, siguiendo la descripción que realizó, podría pasar que se agregue algún valor mayor a 5 s en la columna de tiempos.

Para el Algoritmo 2 esperaría una descripción como la siguiente:

En primer lugar debo poner los primeros valores en la tabla, que son 0 en la de tiempo y 8 en la de posición porque es la posición

inicial. Luego debo agregar un valor en la columna de tiempos, que sea el anterior + 0,5 s, y este paso se debe repetir hasta llegar a 5 s. Posteriormente, en la columna de posiciones debo agregar el valor de la posición anterior + 1,5 m, porque es lo que se desplaza la persona en 1s, y este paso lo debo repetir hasta tener la misma cantidad de valores que en la columna de tiempos.

Los errores que pueden cometer los estudiantes son muy similares a los que se pueden cometer en el otro algoritmo, por lo que no los mencionaré nuevamente. Sin embargo, al tener que realizar dos bucles diferentes puede que surjan nuevos.

Supongamos que un estudiante estableció la condición correcta para terminar el primer bucle (si no lo hace habría que realizar las mismas preguntas que con el Algoritmo 1). En ese caso, puede que quiera imponer la misma condición en el segundo bucle, que es repetir hasta que el último valor de la secuencia de tiempos sea 5 s o más, esperando que la cantidad de repeticiones sea la misma que en el bucle anterior. Este algoritmo fallaría ya que la secuencia de tiempos estaría completa al comenzar este bucle, por lo que no se agregaría ningún valor en la secuencia de posiciones. En consecuencia, se le debería preguntar cuál es el último valor que agregó en la secuencia de tiempos antes de comenzar a agregar valores a la de posiciones, y si esto influye en algo en la condición establecida para terminar de repetir las acciones.

Luego de realizar todas estas preguntas (si fue necesario) para orientar a los estudiantes a realizar la descripción correcta, se esperaría que lleguen a una descripción parecida a las planteadas como correctas. Todo el proceso descrito debería inducir a los estudiantes a tomar conciencia de la coordinación de acciones realizadas ($P \rightarrow C$) y de los cambios sobre los objetos manipulados ($P \rightarrow C'$).

3.1.2. Diseño de la segunda parte (inter→trans)

La transición de la etapa inter a la trans comienza con la elaboración de un pseudocódigo a partir de la descripción en lenguaje natural. Un ejemplo de pseudocódigo correcto para el Algoritmo 1 es:

```
1 t_i = 0
2 t_f = 5
```

```

3| dt = 0.5
4| v = 3
5| x_i = 8
6|
7| t = secuencia con el primer valor de tiempo (0)
8| x = secuencia con el primer valor de posición (8)
9|
10| repetir mientras el ultimo valor de t sea menor que t_f:
11|     agregar a t el valor anterior + dt
12|     agregar a x el valor anterior + v*dt

```

Al realizar este pseudocódigo se pueden cometer varios errores, que podemos vincular a los posibles errores de los estudiantes durante la actividad anterior. Por ejemplo, no comenzar las tablas con los valores iniciales de tiempo y posición se vincula con no declarar estas variables al comienzo del pseudocódigo o con una primera asignación vacía en lugar de con los valores iniciales.

La decisión de indicar que el proceso se repite diez veces, en lugar de hacerlo hasta alcanzar el tiempo final, puede trasladarse directamente al pseudocódigo en la línea donde se inicia el bucle. Si bien esta formulación produce un resultado correcto para el conjunto de valores considerado, su validez se limita a este caso particular y no refleja de manera general las acciones realizadas al elaborar la tabla.

Del mismo modo, el caso de modificar los valores de t y x en lugar de agregar nuevos tiene su equivalente en el pseudocódigo. En este caso, el problema aparece cuando se cambia únicamente el último valor de la secuencia o cuando se declaran t y x como números en vez de secuencias.

Un error que puede aparecer en el pseudocódigo es establecer que el bucle termina si el último valor de la secuencia de tiempos es menor o igual que el tiempo final en vez de solamente menor. Probablemente este error surja al “traducir” que se repite hasta que se cumple una condición (en la descripción de acciones) a decir que se repite mientras se cumple otra condición (en el pseudocódigo). Sin embargo, esta traducción se puede realizar de forma correcta y que el error provenga de establecer que se realizan repeticiones hasta que el último valor de la secuencia de tiempos sea mayor al tiempo final en lugar de mayor o igual. Este error podría surgir al elaborar un programa en vez de el pseudocódigo, ya que en el pseudocódigo se podría seguir utilizando la condición establecida al describir las acciones. Sin embargo, esto ya no es posible en un programa en Python a menos que se utilice un operador *not*,

pero es poco probable que sea la primera opción de un estudiante.

El pseudocódigo para el Algoritmo 2 sería igual, pero cambiando las últimas 3 líneas por:

```
1 repetir mientras el ultimo valor de t sea menor que t_f:
2     agregar a t el valor anterior + dt
3 repetir mientras la cantidad de elementos de x sea menor que la
4     de t:
5     agregar a x el valor anterior + lo que se desplaza en
6         dt
```

En este caso, el error ya mencionado de poner mal la condición para que termine el bucle se podría trasladar directamente a este pseudocódigo.

Una vez elaborado correctamente el pseudocódigo, los estudiantes deben escribir un programa en Python que genere las listas correspondientes. Una posible implementación del Algoritmo 1 es la siguiente:

```
1 t_i = 0
2 t_f = 5
3 dt = 0.5
4 v = 3
5 x_i = 8
6
7 t = [t_i]
8 x = [x_i]
9
10 while t[-1] < t_f:
11     t.append(t[-1] + dt)
12     x.append(x[-1] + v*dt)
```

Mientras que en el caso del Algoritmo 2 habría que cambiar el final por:

```
1 while t[-1] < t_f:
2     t.append(t[-1] + dt)
3 while len(x) < len(t):
4     x.append(x[-1] + v*dt)
```

Todos los errores que se pueden producir al elaborar el pseudocódigo tienen una correspondencia con errores que se pueden producir al elaborar el programa en Python, y la mayoría tienen una conexión muy directa ya que en algunos aspectos los códigos son idénticos. A excepción de dos casos, consideraremos que

la correspondencia entre el error en el pseudocódigo y el error en el programa es trivial y no la describiremos en detalle. Sin embargo, se mencionan todas en la Tabla 3.2.

Uno de los dos casos es el error en el comienzo del bucle. En este caso, un estudiante que repite las acciones una cantidad determinada de veces en vez de dejar de repetir cuando la lista de tiempos llegó a su valor final, probablemente haría un *for* en vez de un *while*. Nuevamente, se puede hacer que el programa funcione haciendo el *for* en vez del *while*, pero esto implica realizar el cálculo de la cantidad de repeticiones antes de comenzar el bucle. Además, este código no tendría una correspondencia con las acciones que una persona realiza cuando elabora la tabla ella misma. Cuando uno elabora la tabla no comienza pensando en cuántas filas tendrá la tabla, lo común es ir agregando los valores hasta llegar al valor de tiempo final. Por lo tanto, un programa coherente con las acciones realizadas debería usar un *while* en lugar de un *for*.

En el caso de los estudiantes que consideran a *t* y *x* como números en vez de secuencias, o que cambian solo el valor final de cada una en vez de agregar valores nuevos, probablemente declaren originalmente la lista de tiempos haciendo *t=t_i* y *x=x_i* o dentro del *while* harían *t[-1]+=dt* y *x[-1]+=v*dt*.

Esta actividad se vincula con varias de las prácticas caracterizadas por Weller et al. (2022). Para elaborar el programa, los estudiantes deben descomponer el problema físico en subproblemas, traducir las relaciones de la cinemática a instrucciones en código, construir un algoritmo que modele el movimiento, aplicar la lógica condicional en la implementación del bucle, generar y manipular datos, escribir el programa, ejecutarlo y depurarlo a partir de los errores detectados. Adicionalmente dado que la actividad es grupal las tareas propuestas se resuelven en forma colaborativa.

En la implementación del algoritmo, en la ejecución del programa y en lograr que dé el resultado correcto, el estudiante construye conocimiento sobre los aspectos computacionales del problema (las acciones que la computadora hace y las modificaciones en las estructuras de datos), o sea el pensamiento del estudiante transita desde *newP* (conoce el algoritmo) hacia *newC* y *newC'* (conoce el programa como objeto ejecutable, no solo como texto).

Tabla 3.2: Posibles errores de los estudiantes en cada etapa en la actividad de MRU.

Error en descripción	Error en pseudocódigo	Error en programa
Algoritmos 1 y 2		
No agregar los valores iniciales en la tabla.	No declarar las secuencias de tiempo y posición, declararlas vacías o asignarles valores incorrectos.	No declarar las listas de tiempo y posición, declararlas vacías o asignarles valores incorrectos.
Decir que se cambian los valores en vez de agregar nuevos.	Considerar a t y x como números en vez de secuencias, o cambiar el valor final en vez de agregar uno nuevo.	Declarar a t y x como datos del tipo <i>float</i> o <i>int</i> en vez de listas, o cambiar el valor final en vez de agregar uno nuevo.
Decir cuántas veces se repite el proceso en vez de decir que se para cuando se llega al valor final en la columna de tiempos.	Establecer en el pseudocódigo cuántas repeticiones hay.	Usar un <i>for</i> en vez de un <i>while</i> .
Decir que la repetición de acciones termina cuando el último tiempo de la tabla es mayor al final en vez de mayor o igual.	Terminar el bucle si el último valor de t es menor o igual que el tiempo final en vez de solo menor.	while $t[-1] <= t_f$:
Algoritmo 2		
Pensar que la repetición para las posiciones termina de la misma forma que la de tiempos.	Para finalizar el segundo bucle establecer la misma condición que en el primero.	El mismo que en el pseudocódigo.

3.2. Segunda actividad: Programa para calcular $x(t)$ y $v(t)$ en el MRUA

A diferencia del MRU, en el Movimiento Rectilíneo Uniformemente Acelerado (MRUA) la velocidad varía con el tiempo, por lo que la posición no se obtiene de forma tan directa. Aunque la expresión correspondiente es una función cuadrática, puede aproximarse sin recurrir al cálculo integral adaptando el programa del MRU: basta con considerar que la velocidad permanece constante en intervalos de tiempo muy pequeños y actualizarla paso a paso.

Antes de realizar esta actividad, es necesario trabajar con los estudiantes sobre la elaboración de tablas de tiempo, velocidad y posición haciendo uso

de esta aproximación. En este caso, se elaboró la tabla para dos ejemplos distintos, por lo que al comenzar la actividad la mayoría de ellos eran capaces de realizarla en una situación con valores distintos a los ejemplos anteriores.

Como los estudiantes ya realizaron también la actividad previa, en la que formalizaron el algoritmo con el que elaboraban la tabla de $x(t)$ en un MRU, enfrentar el mismo problema para un MRUA es mucho más sencillo porque la mayoría de los pasos son muy similares. La principal diferencia es que la velocidad ya no es constante, por lo que se debe actualizar su valor en cada repetición del bucle en el que se agregan nuevas posiciones. Dada esta situación, resulta conveniente aprovechar para realizar también una lista en la que se guarden los valores de velocidad, y eventualmente realizar gráficas en las que se analice su comportamiento respecto al tiempo.

De esta manera, el diseño de la actividad es muy similar a la realizada para el MRU. En una primera etapa, los grupos de estudiantes deben elaborar la tabla y explicar cómo la hicieron, y en una segunda etapa debieron elaborar un pseudocódigo que describe el algoritmo y un programa en Python que lo implementa.

3.2.1. Diseño de la primera parte (intra→inter)

En una primera instancia, se les otorga a los estudiantes la siguiente consigna:

Una persona se encuentra caminando en línea recta y con aceleración constante. Su aceleración es $a = 2 \text{ m/s}^2$, su velocidad inicial es $v_i = 5 \text{ m/s}$ y su posición inicial es $x_i = 8 \text{ m}$. Realiza una tabla en la que se indique la velocidad y la posición de la persona para cada valor de tiempo, donde tomes valores de tiempo cada $0,1 \text{ s}$ entre los valores 0 s y 1 s . Para cada intervalo de $0,1 \text{ s}$ debes suponer que la velocidad se mantiene constante.

Para resolver esta consigna los estudiantes deben calcular cada valor de la tabla a partir de los anteriores, a partir de las ecuaciones

$$v_{\text{nueva}} = v_{\text{anterior}} + a \cdot dt \quad (3.1)$$

$$x_{\text{nueva}} = x_{\text{anterior}} + v_{\text{anterior}} \cdot dt \quad (3.2)$$

donde $v_{anterior}$ y $x_{anterior}$ son los valores de velocidad y posición respectivamente en la fila que se encuentra arriba del valor que se quiere calcular (v_{nueva} o x_{nueva}). Utilizando estas expresiones, los estudiantes deberían obtener la Tabla 3.3.

Tabla 3.3: Posición y velocidad en función del tiempo de una persona que se mueve con aceleración constante. Los valores son calculados suponiendo que la velocidad se mantiene constante cada 0,1 s y que luego cambia abruptamente.

Tiempo (s)	Velocidad (m/s)	Posición (m)
0	5,0	8,0
0,1	5,2	8,5
0,2	5,4	9,02
0,3	5,6	9,56
0,4	5,8	10,12
0,5	6,0	10,7
0,6	6,2	11,3
0,7	6,4	11,92
0,8	6,6	12,56
0,9	6,8	13,22
1,0	7,0	13,9

Nuevamente, consideraremos como periferia P el caso en el que los estudiantes logran realizar la tabla, ya sea en un primer intento o mediante la orientación del docente. Esta parte de la actividad se centrará entonces en la conceptualización del algoritmo.

En este problema podemos identificar a 9 magnitudes involucradas. Por un lado, hay 6 parámetros que no caracterizaremos como objetos porque no se modifican durante la resolución, que son la posición inicial (x_i), la velocidad inicial (v_i), la aceleración (a), el tiempo inicial (t_i), el tiempo final (t_f) y el paso dt . Las otras tres, que caracterizaremos como objetos, son la secuencia de tiempos, la de velocidades y la de posiciones.

Las acciones necesarias para resolver este problema son:

- Agregar en las tres secuencias los valores iniciales que indican los parámetros.
- Agregar un valor en la secuencia de tiempos, que sea igual al anterior + dt .
- Agregar un valor en la secuencia de velocidades, que sea igual al anterior + $a \cdot dt$.

- Agregar un valor en la secuencia de posiciones, que sea igual al anterior $+ v_{anterior} \cdot dt$.
- Repetir los tres pasos anteriores hasta que el último valor de la secuencia de tiempos sea mayor o igual al valor final.

Otro camino muy similar para resolverlo es:

- Agregar en las tres secuencias los valores iniciales que indican los parámetros.
- Agregar un valor en la secuencia de tiempos, que sea igual al anterior $+ dt$.
- Repetir el paso anterior hasta que el último valor de la secuencia de tiempos sea mayor o igual al valor final.
- Agregar un valor en la secuencia de velocidades, que sea igual al anterior $+ a \cdot dt$.
- Repetir el paso anterior hasta que la secuencia de velocidades tenga la misma cantidad de elementos que la de los tiempos.
- Agregar un valor en la secuencia de posiciones, que sea igual al anterior $+ v_{anterior} \cdot dt$.
- Repetir el paso anterior hasta que la secuencia de posiciones tenga la misma cantidad de elementos que la de los tiempos.

Sin embargo, este nuevo camino es más complejo porque no es sencillo identificar a $v_{anterior}$ cuando se elabora la secuencia de posiciones. Esto es porque en este algoritmo se hace la secuencia de tiempos de forma completa, luego la de velocidades y finalmente la de posiciones, en lugar de ir haciendo las tres a la vez. Si bien al hacerlo a mano con una tabla es fácil identificar a $v_{anterior}$ porque es la que se encuentra en la fila de arriba, cuando se pase a la siguiente etapa y se deba realizar un programa que ejecute estas acciones ya no será tan sencillo. Por lo tanto, en esta actividad vemos conveniente alentar a los estudiantes a realizar su resolución mediante el primer algoritmo y no con este, y por este motivo a partir de ahora no consideraremos más esta solución.

Una vez completada la tarea, los estudiantes deben explicar el procedimiento que siguieron. A continuación se presenta un ejemplo de descripción considerada correcta:

En primer lugar debemos poner los primeros valores en la tabla, que son 0 en el tiempo, 5 en la velocidad y 8 en la posición. Luego

debemos agregar un valor en la columna de tiempos, que sea el anterior $+ 0,1\text{ s}$, en la columna de velocidades debemos agregar el valor de la velocidad anterior $+ a \cdot dt$ y en la columna de posiciones debemos agregar el valor de la posición anterior $+ v_{anterior} \cdot dt$. Estos tres pasos los debemos repetir hasta llegar a 1 s en la columna de los tiempos.

Es posible que en la primera descripción tengan algunos errores, por lo que el docente debería orientarlos con alguna preguntas hasta llegar a una descripción correcta. La mayoría de estos errores son los mismos que en la actividad del MRU y se presentan en la Tabla 3.4.

De acuerdo a la ley general de Piaget (ver sección 3.1.2), una vez que los estudiantes dan una explicación correcta del algoritmo, toman conciencia de la coordinación de acciones realizadas ($P \rightarrow C$) y de los cambios sobre los objetos manipulados ($P \rightarrow C'$), dado que la explicación coincide con lo que efectivamente hacen para solucionar el problema. Es en esa interacción entre P y los centros C y C' que el estudiante construye conocimiento conceptual sobre el algoritmo empleado..

3.2.2. Diseño de la segunda parte (inter→trans)

Para iniciar la transición de la etapa inter a la trans, los estudiantes deben elaborar un pseudocódigo a partir de la descripción realizada en lenguaje natural. Un ejemplo de pseudocódigo correcto es:

```

1 t_i = 0
2 t_f = 5
3 dt = 1
4 a = 2
5 v_i = 3
6 x_i = 8
7
8 t = secuencia con el primer valor de tiempo (0)
9 v = secuencia con el primer valor de velocidad (5)
10 x = secuencia con el primer valor de posición (8)
11
12 repetir mientras el ultimo valor de t sea menor que t_f:
13     agregar a t el valor anterior + dt
14     agregar a v el valor anterior + a*dt

```

15

```
agregar a x el valor anterior + v_anterior*dt
```

Nuevamente, los errores que se pueden llegar a cometer son los mismos que en la actividad del MRU y se presentan en la Tabla 3.4, por lo que no los volvemos a analizar en detalle. Una vez elaborado el pseudocódigo, los estudiantes deben escribir un programa en Python que genere las listas correspondientes. Una posible implementación es la siguiente:

```

1 t_i = 0
2 t_f = 5
3 dt = 0.1
4 a = 2
5 v = 5
6 x_i = 8
7
8 t = [t_i]
9 v = [v_i]
10 x = [x_i]
11
12 while t[-1] < t_f:
13     t.append(t[-1] + dt)
14     v.append(v[-1] + a*dt)
15     x.append(x[-1] + v[-2]*dt)

```

Si bien en este programa la mayoría de los errores que pueden cometer los estudiantes son muy similares a los del MRU, hay uno nuevo que es específico de este nuevo problema. Al calcular las posiciones nuevas utilizando el valor anterior de velocidad, se puede llegar a cometer un error al tratar de identificar ese valor. En la última línea del programa se hace referencia a él poniendo $v[-2]$, que es el penúltimo valor de la lista v , ya que al haber agregado en la línea anterior el valor nuevo de velocidad lo que se debe tomar como referencia para calcular la posición es la penúltima velocidad en lugar de la última. Sin embargo, muchos estudiantes suelen poner $v[-1]$ porque piensan que deben utilizar el último valor en lugar del penúltimo.

En lugar de utilizar el penúltimo valor de la lista de velocidades, se puede invertir el orden de las últimas dos líneas del programa y utilizar la última velocidad. Es decir, si en primer lugar se agrega la posición nueva utilizando la última velocidad y luego se agrega la velocidad nueva, se obtiene lo mismo que con el otro programa. De esta manera, el bucle quedaría de la siguiente

Tabla 3.4: Posibles errores de los estudiantes en cada etapa en la actividad de MRUA.

Error en descripción	Error en pseudocódigo	Error en programa
No agregar los valores iniciales de tiempo, velocidad y posición.	No declarar las secuencias t, v y x, o declararlas vacías.	No declarar las listas t, v y x, o declararlas vacías.
Decir que se cambian los valores en vez de agregar nuevos.	Considerar a t, v y x como números en vez de secuencias, o cambiar el valor final en vez de agregar uno nuevo.	Declarar a t, v y x como datos del tipo <i>float</i> o <i>int</i> en vez de listas, o cambiar el valor final en vez de agregar uno nuevo.
Confundir la velocidad a usar para calcular la posición (usar la nueva en lugar de la anterior).	Usar la velocidad actual en vez de la anterior en la ecuación de posición.	Escribir x.append(x[-1] + v[-1]*dt) en vez de x.append(x[-1] + v[-2]*dt).
Decir cuántas veces se repite el proceso en vez de decir que se para cuando se llega al valor final en la columna de tiempos.	Establecer en el pseudocódigo cuántas repeticiones hay.	Usar un <i>for</i> en vez de un <i>while</i> .
Decir que la repetición de acciones termina cuando el último tiempo de la tabla es mayor al final en vez de mayor o igual.	Terminar el bucle si el último valor de t es menor o igual que el tiempo final en vez de solo menor.	while t[-1] <= t_f:

forma:

```

1 while t[-1] < t_f:
2     t.append(t[-1] + dt)
3     x.append(x[-1] + v[-1]*dt)
4     v.append(v[-1] + a*dt)

```

Al igual que en la actividad anterior, en esta se ponen en juego las mismas prácticas descritas por Weller et al. (2022): descomposición del problema, traducción entre representaciones físicas y computacionales, construcción y depuración de algoritmos, manipulación de datos y aplicación de la lógica computacional. Sin embargo, al enfrentarse a una tarea de naturaleza similar pero con mayor complejidad conceptual y computacional, aparece además la práctica de agregar complejidad al modelo. En este caso, los estudiantes deben extender el modelo del MRU incorporando la variación de la velocidad, lo

que implica reconocer nuevas relaciones entre variables y adaptar el algoritmo previo a un sistema de mayor complejidad.

De acuerdo a la ley de la cognición extendida (ver sección 2.2.3), una vez que los estudiantes construyen y ejecutan un programa correcto, toman conciencia de las acciones realizadas por la computadora ($newP \rightarrow newC$) y de los cambios que las mismas producen en las estructuras de datos ($newP \rightarrow newC$), donde se comprenden los aspectos computacionales que no necesariamente aparecen en el algoritmo. Es en esa interacción entre $newP$ y los centros $newC$ y $newC'$ que el estudiante construye conocimiento conceptual sobre el programa como objeto ejecutable.

3.3. Tercera actividad: Análisis de movimientos rectilíneos

Los programas elaborados en las actividades previas son especialmente útiles para elaborar gráficas de $x(t)$ y de $v(t)$ en movimientos rectilíneos. Al haber realizado las listas con estas magnitudes, se pueden utilizar de forma directa para elaborar las gráficas. Además, solo con modificar los parámetros del problema al principio del programa se pueden realizar estas gráficas para cualquier MRU o MRUA, y no solo para los ejemplos trabajados en esas actividades.

No solo es posible construir con facilidad las gráficas de posición y velocidad en función del tiempo, sino que las listas elaboradas pueden aprovecharse para generar visualizaciones más elaboradas que permitan observar simultáneamente el movimiento y sus representaciones. Con este propósito, se desarrolló un simulador en Python especialmente para este trabajo, que permite visualizar a una persona desplazándose mientras las gráficas de $x(t)$ y $v(t)$ se construyen en tiempo real. El simulador utiliza como datos de entrada las listas de tiempo, velocidad y posición elaboradas por los estudiantes, y está diseñado para trabajar con posiciones comprendidas entre -100 m y 100 m, tomando la posición 0 como el centro de la pantalla. La Figura 3.1 muestra una captura de pantalla correspondiente a una de las simulaciones realizadas.

En esta actividad se proporciona a los estudiantes un archivo con el simulador desarrollado, y ellos deben utilizarlo para identificar características de los movimientos rectilíneos a partir de simulaciones con distintas condiciones iniciales.

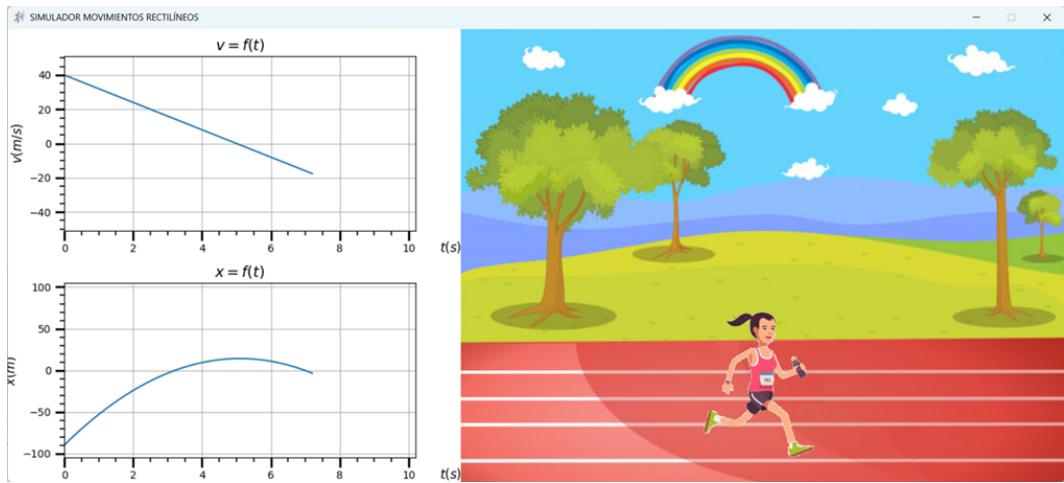


Figura 3.1: Simulación realizada con $x_i = -90\text{ m}$, $v_i = 40\text{ m/s}$ y $a = -8\text{ m/s}^2$. A la izquierda se construyen las gráficas en tiempo real y a la derecha se mueve una persona. La parte izquierda de la pista corresponde a la posición -100 m , el centro a 0 m y la derecha a 100 m .

Para identificar las características de los movimientos se les otorga a los estudiantes algunas condiciones iniciales a utilizar y preguntas a responder en base a la simulación que obtengan, así ven todas las opciones posibles de lo que puede ocurrir en un MRU o en un MRUA y no les queda ninguna opción sin ver. En base a esas preguntas los estudiantes deben identificar

- Qué ocurre cuando la posición de la persona es negativa y qué ocurre cuando es positiva.
- En qué casos la persona se mueve hacia la izquierda y en qué casos se mueve hacia la derecha.
- Qué vínculo hay entre la inclinación de la curva de la gráfica de $x(t)$ y la velocidad.
- Qué ocurre con la velocidad cuando la aceleración es negativa y qué ocurre cuando es positiva.
- Qué debe ocurrir para que la persona frene.

En la segunda parte de la actividad, los estudiantes interactúan con un videojuego desarrollado en el marco de este trabajo, también programado en Python, en el que deben reproducir gráficas de $x(t)$. En este entorno se presenta una gráfica de posición en función del tiempo y los estudiantes deben mover, con el ratón, a una persona de manera que su desplazamiento coincida con el representado en la gráfica. Mientras realizan el movimiento, el programa genera

en tiempo real una nueva gráfica de $x(t)$ que se superpone con la original, lo que permite comparar ambas representaciones y analizar las correspondencias entre el movimiento efectuado y el modelo gráfico. A partir de la distancia media entre la gráfica obtenida y la solicitada, el programa calcula una puntuación. La Fig. 3.2 muestra capturas de pantalla del videojuego.



Figura 3.2: Videojuego de imitación de gráficas. En la imagen superior izquierda se muestra la gráfica que el jugador debe replicar, moviendo a la persona con el ratón. La imagen superior derecha corresponde a una captura durante el movimiento, donde la gráfica azul representa lo que se debe reproducir y la naranja lo realizado hasta ese momento. En la imagen inferior izquierda se observa la puntuación obtenida tras completar la tarea, y en la inferior derecha una simulación que compara el movimiento esperado con el efectuado por el jugador.

El videojuego incluye tres niveles de dificultad. En el nivel fácil se genera aleatoriamente una gráfica correspondiente a un MRU; en el nivel intermedio, una de un MRUA; y en el nivel difícil se combinan intervalos que alternan ambos tipos de movimiento y momentos de reposo. La Fig. 3.3 muestra ejemplos de las gráficas generadas en cada nivel.

En la primera instancia, los estudiantes jugaron utilizando el ratón para controlar el movimiento de la persona en la pantalla. En una segunda etapa,

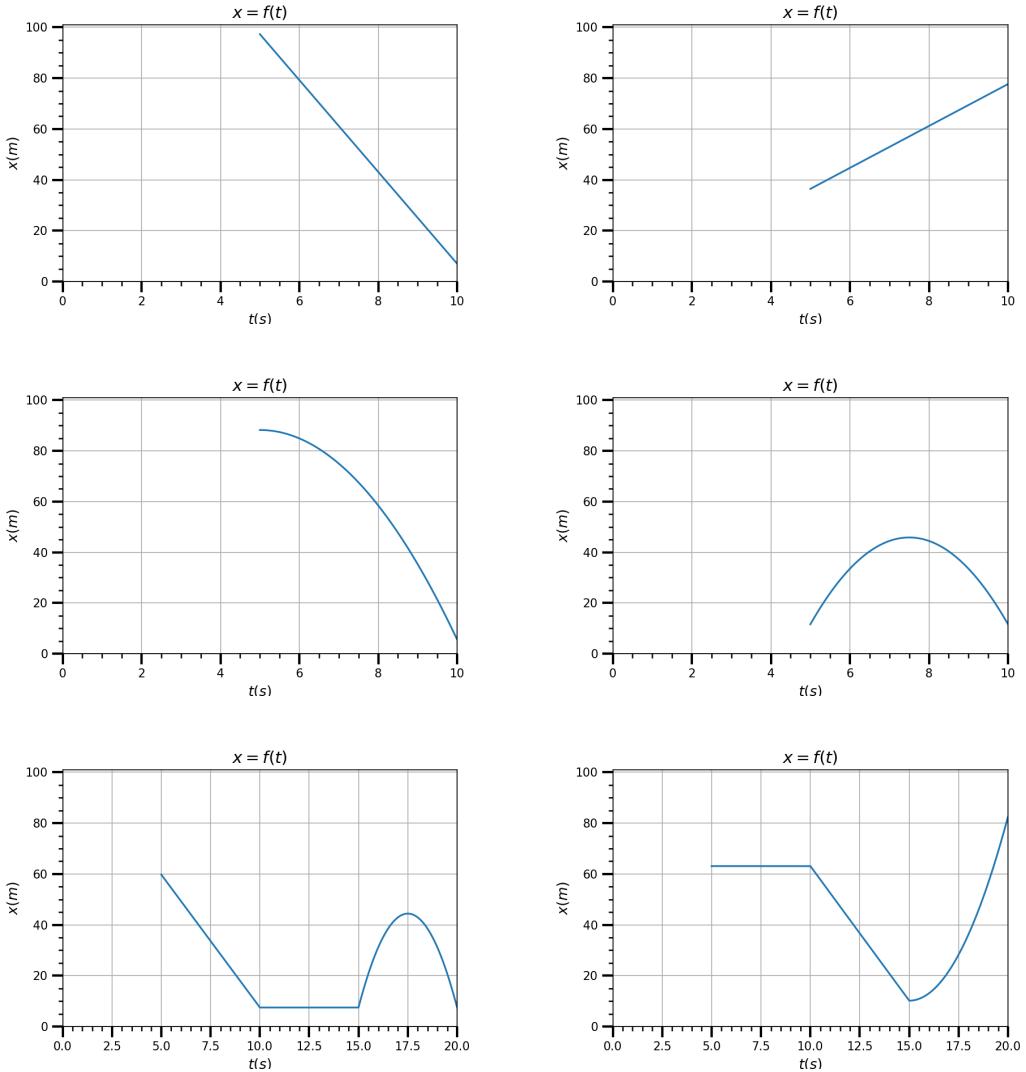


Figura 3.3: Gráficas generadas en cada nivel del videojuego. En cada nivel de dificultad, las gráficas se generan aleatoriamente. Las imágenes de la fila superior corresponden al nivel fácil, las de la fila central al nivel medio y las de la inferior al nivel difícil.

realizaron la misma actividad, pero moviéndose ellos mismos frente a un sensor ultrasónico *HC-SR04* conectado a una placa *Arduino*, que registraba en tiempo real la posición de su cuerpo y la enviaba a la computadora. Si bien el código para la lectura del sensor podría haber sido elaborado por los propios estudiantes, en este caso se les proporcionó una placa con un programa ya subido, de modo que solo debían conectar el dispositivo y ejecutar el juego. En esta versión de la actividad, además de jugar, debían bosquejar las gráficas de $v(t)$ y $a(t)$ antes de intentar replicar la gráfica de $x(t)$, con el fin de anticipar

las variaciones de velocidad y aceleración esperadas.

Esta actividad incorpora además nuevas prácticas de pensamiento computacional descritas por Weller et al. (2022). En primer lugar, la práctica de generalización, dado que los estudiantes reutilizan los programas elaborados previamente para analizar nuevos casos de MRU y MRUA, modificando solo los parámetros iniciales. En segundo lugar, la práctica de análisis de datos, que se pone en juego al interpretar la información obtenida a partir de las simulaciones y de las representaciones gráficas generadas en tiempo real. Estas acciones se articulan con estrategias que han demostrado favorecer el aprendizaje de la cinemática, como la visualización simultánea de los movimientos y sus gráficas (Sokoloff et al. 2007). Asimismo, el uso de la ludificación mediante el videojuego introduce un componente motivacional que promueve la exploración activa.

Capítulo 4

Características de los aprendizajes en cinemática

Este capítulo analiza las características de los aprendizajes desarrollados por los estudiantes durante la implementación de la secuencia didáctica. El objetivo es describir y comprender cómo evolucionaron sus concepciones sobre los conceptos básicos de cinemática.

Consideramos dos fuentes principales de información. En primer lugar, aplicamos una versión reducida del TUGK en español (Zavala et al. 2019), con el propósito de identificar las concepciones iniciales y finales de los estudiantes sobre los conceptos de posición, velocidad y aceleración, así como su interpretación de representaciones gráficas. El test lo administramos a $N = 78$ estudiantes, distribuidos en tres grupos de clase, en dos instancias: antes y después de la implementación de la secuencia didáctica. Consideramos únicamente a los estudiantes que participaron en ambas instancias, descartando los casos incompletos.

En segundo lugar, analizamos una actividad centrada en la interpretación y elaboración de gráficas de un movimiento, incorporada en la secuencia didáctica. Esta actividad complementa los resultados obtenidos a partir del test, ya que permite observar con mayor detalle cómo los estudiantes aplicaron sus conocimientos de cinemática al interpretar y relacionar diferentes representaciones de un mismo fenómeno de movimiento.

4.1. Evaluación mediante TUGK

4.1.1. Selección de preguntas del test

El test original está compuesto por 26 ítems distribuidos en siete objetivos. Dado que su extensión resulta poco adecuada para estudiantes de 14 a 15 años, seleccionamos los tres objetivos que consideramos más pertinentes para esta investigación y utilizamos los ítems correspondientes. Además, cabe señalar que el TUGK fue diseñado originalmente para estudiantes universitarios, por lo que no está específicamente orientado al nivel de K9. Sin embargo, dado que los conceptos que evalúa, como posición, velocidad, aceleración e interpretación de gráficas, son precisamente los que buscamos analizar en este trabajo, consideramos pertinente emplear una versión adaptada. Esta selección busca preservar la validez conceptual del instrumento y, al mismo tiempo, adecuar su extensión y nivel de complejidad a las características de la población participante.

Los objetivos no seleccionados abordan contenidos afines, pero se centran en el cálculo de magnitudes físicas más que en el análisis cualitativo de los movimientos. En los ítems asociados a estos objetivos se solicita determinar la velocidad en un punto a partir de la gráfica de posición en función del tiempo, la aceleración en un punto a partir de la gráfica de velocidad en función del tiempo, el desplazamiento a partir de un intervalo en la gráfica de velocidad en función del tiempo y la variación de velocidad en un intervalo a partir de la gráfica de aceleración en función del tiempo.

Si bien el criterio principal para la selección de los ítems fue su contenido conceptual, también revisamos los índices de dificultad, discriminación y punto biserial de cada pregunta, con el propósito de garantizar la validez del instrumento tras la eliminación parcial de ítems. Los valores de estos índices para cada pregunta se presentan en la Tabla 4.1, y fueron obtenidos por Zavala et al. (2017) a partir de la aplicación del test a 471 estudiantes de una universidad privada en México.

El **índice de dificultad** P expresa el grado de complejidad de un ítem y se define como el cociente entre el número de respuestas correctas y el total de respuestas:

$$P = \frac{N_{\text{correctas}}}{N} \quad (4.1)$$

Este índice toma valores entre 0 y 1: vale 0 cuando todas las personas responden

Tabla 4.1: Índices de dificultad (P), discriminación (D) y punto biserial (r) por ítem, con descripción original y agrupados por objetivo. El número entre paréntesis corresponde al ítem en el test original.

Pregunta	Descripción del ítem	P	D	r
Objetivo 1: Seleccionar gráfica a partir de otra				
P3 (11)	Determinar la gráfica correspondiente que implica “un paso hacia adelante”.	0.45	0.66	0.51
P5 (14)	Determinar la gráfica correspondiente que implica “un paso hacia atrás”.	0.54	0.71	0.55
P9 (21)	Determinar la gráfica correspondiente que implica “un paso hacia atrás”.	0.50	0.60	0.44
P6 (15)	Determinar la gráfica correspondiente que implica “un paso hacia atrás”.	0.46	0.64	0.50
Objetivo 2: Describir movimiento a partir de una gráfica				
P2 (8)	Gráfica de posición: determinar movimiento particular.	0.51	0.72	0.58
P1 (3)	Gráfica de posición: determinar movimiento de velocidad constante.	0.58	0.63	0.51
P11 (24)	Gráfica de velocidad: determinar movimiento de aceleración constante.	0.32	0.71	0.60
P7 (17)	Gráfica de velocidad: determinar posición que incrementa uniformemente.	0.70	0.71	0.59
P12 (25)	Gráfica de aceleración: determinar velocidad que incrementa uniformemente.	0.54	0.78	0.61
Objetivo 3: Seleccionar gráfica a partir de descripción				
P4 (12)	Velocidad constante: identificar gráficas.	0.48	0.80	0.63
P10 (22)	Aceleración constante: identificar gráficas.	0.48	0.73	0.58
P13 (26)	Velocidad que se incrementa uniformemente: identificar gráficas.	0.51	0.83	0.65
P8 (20)	Aceleración que se incrementa uniformemente: identificar gráficas.	0.52	0.43	0.32

de forma incorrecta y 1 cuando todas lo hacen correctamente.

El **índice de discriminación** D indica en qué medida una pregunta distingue entre estudiantes con alto y bajo dominio de los conceptos evaluados. Para calcularlo, se divide la muestra en dos grupos de igual tamaño: uno con los puntajes más altos y otro con los más bajos. Para un ítem dado, se contabilizan las respuestas correctas en cada grupo, N_H y N_L , y se define:

$$D = \frac{N_H - N_L}{N/2} \quad (4.2)$$

donde N representa el número total de estudiantes. El valor de D varía entre -1 y $+1$. Un valor alto indica que los estudiantes con mejores resultados tienden a responder correctamente el ítem, mientras que quienes obtienen puntajes más bajos suelen fallarlo. En cambio, valores bajos pueden deberse a que la pregunta resulta demasiado fácil (la mayoría responde correctamente), demasiado difícil (la mayoría responde incorrectamente) o presenta un comportamiento anómalo, en el que los estudiantes con bajo rendimiento aciertan con mayor frecuencia que los de alto rendimiento (Ding et al. 2006).

El **índice de punto biserial** r_{pb} mide la consistencia de cada ítem con el test completo. A diferencia del índice de discriminación, que compara explícitamente a los estudiantes de alto y bajo desempeño, el punto biserial cuantifica la correlación entre la respuesta a un ítem (correcta o incorrecta) y la puntuación total obtenida en el test. Se define como

$$r_{pb} = \frac{\bar{X}_1 - \bar{X}}{\sigma_X} \sqrt{\frac{P}{1 - P}} \quad (4.3)$$

donde \bar{X}_1 representa el puntaje promedio de quienes respondieron correctamente el ítem, \bar{X} el promedio general del test, σ_X la desviación estándar de los puntajes totales y P el índice de dificultad del ítem. El valor de este índice varía entre -1 y $+1$. Los valores positivos altos indican que los estudiantes con mejor desempeño global tienden a responder correctamente el ítem, lo que muestra una buena coherencia con el resto del test. En cambio, los valores cercanos a cero señalan una relación débil con el desempeño general, mientras que los valores negativos sugieren que el ítem podría estar mal redactado o evaluar un concepto distinto del abordado por la prueba en su conjunto (Ding et al. 2006).

Las trece preguntas seleccionadas presentan índices de discriminación supe-

riores a 0.4 (siendo el menor $D = 0.43$, y el resto $D \geq 0.60$) y valores de punto biserial mayores a 0.30. En conjunto, conforman un grupo de ítems con buena capacidad de diferenciación entre estudiantes de distinto nivel de desempeño y con adecuada consistencia interna respecto del test completo. Estos valores superan los criterios comúnmente aceptados en PER, que establecen $D \geq 0.3$ y $r_{pb} \geq 0.2$ como umbrales de calidad para la validez de un ítem (Ding et al. 2006; Doran, 1980; Zavala et al. 2017). Por lo tanto, consideramos razonable utilizar esta selección de preguntas en lugar del test completo.

4.1.2. Aprendizajes en cada categoría

En esta sección se presentan los resultados obtenidos por los estudiantes en cada uno de los ítems del test. Para cuantificar los cambios conceptuales en cada ítem se utilizó la d de Cohen (2013), definida como

$$d = \frac{\bar{x}_{post} - \bar{x}_{pre}}{s} \quad (4.4)$$

donde \bar{x}_{post} es la fracción de respuestas correctas en el postest, \bar{x}_{pre} la fracción correspondiente al pretest y s el desvío estándar, calculado a partir de

$$s = \sqrt{\frac{\bar{x}_{pre}(1 - \bar{x}_{pre}) + \bar{x}_{post}(1 - \bar{x}_{post})}{2}} \quad (4.5)$$

lo que corresponde a la varianza binomial esperada para proporciones dicotómicas (Agresti y Finlay, 2009).

La d de Cohen representa una diferencia media estandarizada, es decir, el número de desvíos estándar que separa el promedio del pretest del promedio del postest. Por ejemplo, un valor de $d = 0,5$ indica que el desempeño en el postest se encuentra medio desvío estándar por encima del obtenido en el pretest. En términos generales, se considera que $d = 0,2$ representa un cambio pequeño, $d = 0,5$ un cambio moderado y $d = 0,8$ un cambio grande (Cohen, 2013).

A continuación se presentan los resultados organizados por categoría conceptual (correspondencia entre gráficas del movimiento, describir un movimiento a partir de una gráfica y seleccionar gráficas a partir de descripciones). En cada ítem se reportan la fracción de aciertos pre y post, el tamaño del efecto d de Cohen para proporciones y un análisis cualitativo centrado en las

concepciones alternativas más frecuentes.

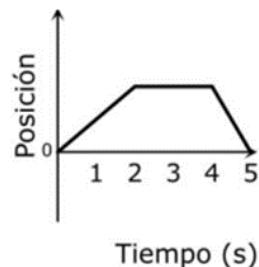
4.1.2.1. Correspondencia entre gráficas del movimiento

En esta categoría de preguntas a los estudiantes se les otorga una gráfica y deben seleccionar otra que corresponda al mismo movimiento. En las preguntas 3 y 5 se evalúa el concepto de pendiente, ya que a partir de una gráfica de posición en función del tiempo deben elegir la correspondiente de velocidad, o a partir de la de velocidad deben elegir la de aceleración. En la 9 y en la 6 se realiza lo contrario, a partir de una gráfica de velocidad deben identificar la de posición y a partir de la de aceleración deben seleccionar al de velocidad, por lo que se evalúa el entendimiento del área bajo la curva.

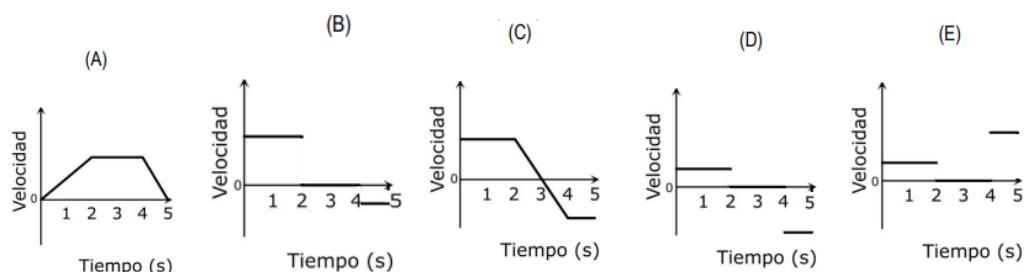
En esta sección se presentan las cuatro preguntas y los resultados obtenidos por los estudiantes.

Pregunta 3

La gráfica muestra la posición en función del tiempo de un objeto durante un intervalo de 5 s.

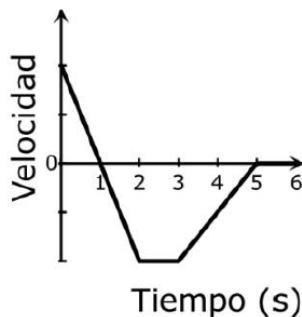


¿Cuál de las siguientes gráficas de velocidad en función del tiempo representaría mejor el movimiento del objeto durante dicho intervalo?

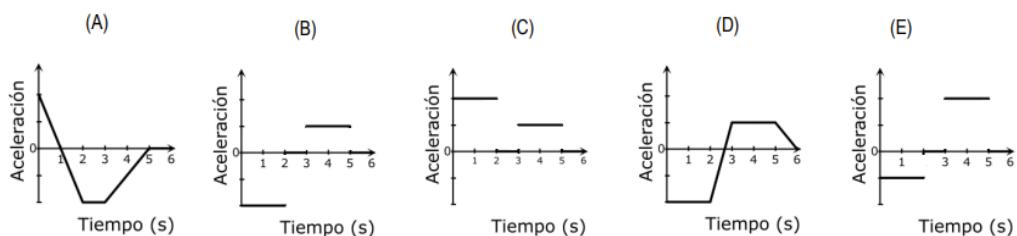


Pregunta 5

La gráfica de abajo muestra la velocidad de un objeto en un intervalo de 6 s.

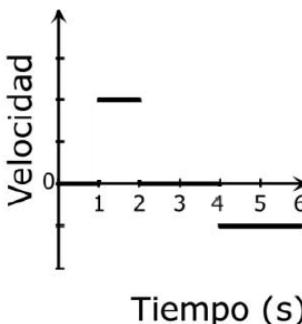


¿Cuál de las siguientes gráficas de aceleración en función del tiempo representaría mejor el movimiento del objeto durante el mismo intervalo?

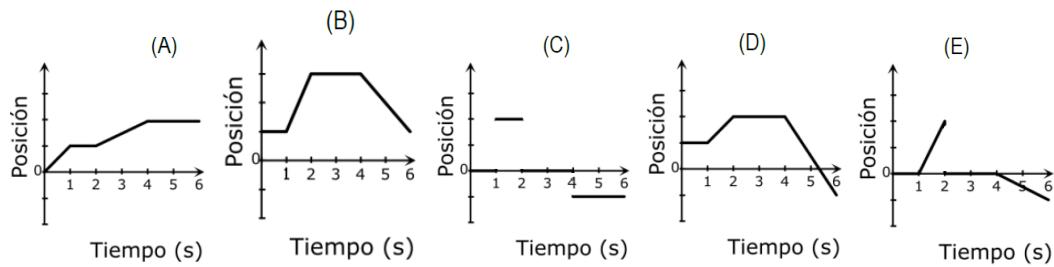


Pregunta 9

La gráfica de abajo muestra la velocidad en función del tiempo para el movimiento de un objeto en un intervalo de 6 s.

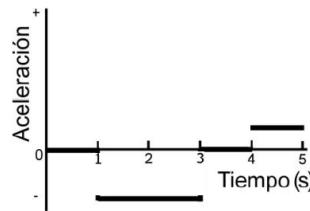


¿Cuál de las siguientes gráficas de posición en función del tiempo representaría mejor el movimiento del objeto durante dicho intervalo de tiempo?

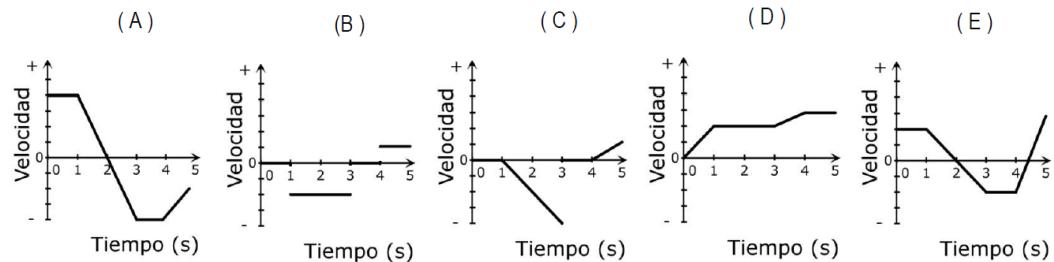


Pregunta 6

La gráfica de abajo muestra la aceleración en función del tiempo para el movimiento de un objeto en un intervalo de 5 s.



¿Cuál de las siguientes gráficas de velocidad en función del tiempo representaría mejor el movimiento del objeto durante dicho intervalo de tiempo?



Resultados

Los resultados obtenidos por los estudiantes en esta categoría se encuentran en la Tabla 4.2.

En la pregunta 3, los estudiantes seleccionaron mayoritariamente la opción A en el pretest, correspondiente a una gráfica de velocidad en función del tiempo con la misma forma que la de posición en función del tiempo presentada. En el postest, solo cinco estudiantes mantuvieron esta elección, mientras que las opciones B y D, siendo esta última la correcta, mostraron un incremento significativo en su frecuencia de selección. Aunque la opción B no es estrictamente correcta, el aumento de respuestas en esta categoría puede interpretarse

Tabla 4.2: Resultados en seleccionar gráfica a partir de otra. La tabla contiene la cantidad de estudiantes que eligió cada opción dentro de cada pregunta, y se marca en negrita cuál es la opción correcta. También se indica la d de Cohen para cada una de las preguntas.

Pregunta 3		Pregunta 5		Pregunta 9		Pregunta 6	
Pre test	Post test						
A	41	5	32	7	7	2	9 25
B	8	32	15	46	15	22	25 9
C	12	7	9	5	18	5	16 14
D	11	31	15	7	21	32	12 9
E	6	3	7	13	17	17	16 21
d	0,60		0,89		0,21		0,51

como un avance conceptual parcial. En dicha opción, la forma general de la gráfica coincide con la correcta, pero el valor absoluto de la velocidad en el primer intervalo es mayor que en el tercero, cuando debería ser menor. Esto sugiere que los estudiantes que eligieron B comprendieron la estructura general del movimiento: reconocieron que en el primer intervalo la velocidad es constante y positiva, en el segundo nula y en el tercero constante y negativa, aunque erraron en la comparación de magnitudes. Si se agrupan las respuestas B y D como conceptualmente consistentes, el tamaño del efecto conjunto sería $d = 1,37$. Si bien este ajuste no modifica el resultado principal, dado que la opción D es la única correcta, sugiere que la ganancia cualitativa podría ser mayor que la reflejada por el valor original de d . En conjunto, los resultados evidencian un progreso notable en la capacidad de los estudiantes para identificar la velocidad de un objeto a partir de la gráfica de su posición.

En la pregunta 5, el valor de $d = 0,89$ indica una ganancia conceptual considerable. En el pretest, la mayoría de los estudiantes había seleccionado la opción A, que mostraba una gráfica de aceleración con la misma forma que la de velocidad otorgada. En el postest, la frecuencia de esta elección disminuyó de manera marcada. Esta pregunta también incluye una opción de respuesta parcialmente correcta, la E, en la cual los estudiantes reconocen que en el primer intervalo la aceleración es constante y negativa, en el segundo nula y en el tercero constante y positiva, aunque sin comparar adecuadamente los valores absolutos de las aceleraciones en el primer y tercer intervalo. Al realizar nuevamente el análisis bajo un supuesto alternativo, considerando la opción E

como conceptualmente correcta, se obtiene un tamaño del efecto de $d = 1,08$, lo que sugiere que la mejora conceptual fue ligeramente superior a la reflejada por el valor original de d .

En la pregunta 9, los resultados muestran un cambio conceptual menor que en los ítems anteriores. Sin embargo, la opción D, que fue la más seleccionada por los estudiantes, puede considerarse parcialmente correcta. Quienes eligieron esta alternativa reconocieron que la posición permanece constante cuando la velocidad es nula, que aumenta uniformemente cuando la velocidad es constante y positiva, y que disminuye uniformemente cuando la velocidad es constante y negativa. No obstante, no identificaron que la inclinación de la gráfica de posición aumenta a medida que lo hace el valor absoluto de la velocidad. Si se considera esta respuesta como correcta, el tamaño del efecto sería $d = 0,48$, lo que indica un cambio de magnitud moderada, superior al que sugiere el valor original de d . Por otra parte, la opción C, que puede interpretarse como la menos consistente conceptualmente, ya que refleja la idea de que la gráfica de velocidad en función del tiempo tiene la misma forma que la de posición en función del tiempo, mostró una disminución considerable en su frecuencia de selección, pasando de 18 a 5 estudiantes.

En la pregunta 6, el valor de $d = 0,51$ indica un cambio de magnitud intermedia. En este caso, la opción E podría considerarse parcialmente correcta. Al recalcular el tamaño del efecto bajo el supuesto alternativo de incluirla como respuesta válida, el valor se incrementa levemente a $d = 0,56$. Por lo tanto, el cambio conceptual evidenciado en este ítem puede caracterizarse como de magnitud moderada. Nuevamente, la opción con mayor descenso en el número de selecciones fue la considerada menos consistente, la B, que expresa la idea de que las gráficas de velocidad en función del tiempo tienen la misma forma que las de aceleración en función del tiempo.

En promedio, el valor de d de Cohen para esta categoría de preguntas fue $d = 0,56$, y ascendería a $d = 0,87$ bajo los supuestos alternativos analizados en cada caso. Esto indica cambios conceptuales de magnitud intermedia que tienden a ser significativos. Además, en todas las preguntas se observaron los descensos más marcados en las opciones que representaban gráficas de igual forma para magnitudes distintas, lo que puede interpretarse como una reducción de una concepción inicial recurrente. En conjunto, los resultados muestran una mayor consolidación en la comprensión del concepto de pendiente que en la relación con el área bajo la curva.

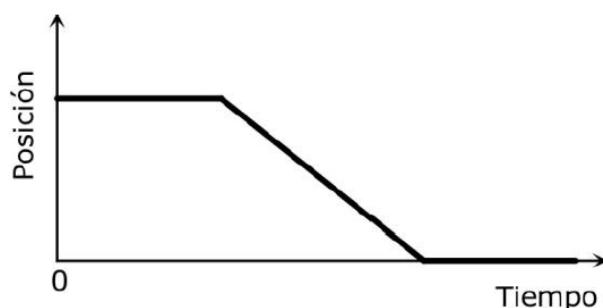
4.1.2.2. Describir un movimiento a partir de una gráfica

En esta categoría de preguntas, los estudiantes reciben una gráfica que representa un movimiento y deben seleccionar la descripción que mejor se corresponde con ella. La pregunta 2 constituye un caso particular: a partir de una gráfica de posición en función del tiempo, los estudiantes deben identificar que el objeto permanece en reposo al inicio, luego se desplaza en sentido opuesto al del eje y finalmente vuelve a detenerse. En las cuatro preguntas restantes, las situaciones proponen reconocer si la velocidad o la aceleración son constantes, o si la posición o la velocidad varían de manera uniforme.

En esta sección se presentan las cinco preguntas junto con los resultados obtenidos por los estudiantes.

Pregunta 2

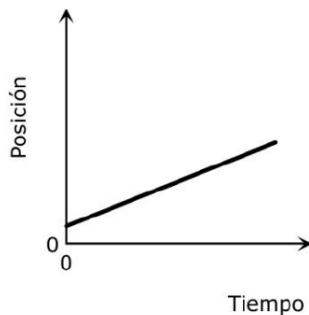
La gráfica muestra la posición en función del tiempo del movimiento de un objeto. ¿Cuál de las frases es la mejor interpretación?



- (A) El objeto rueda a lo largo de una superficie horizontal, después cae rodando por una pendiente y finalmente se detiene.
- (B) El objeto no se mueve al principio, después cae rodando por una pendiente y finalmente se detiene.
- (C) El objeto se mueve a velocidad constante, después frena hasta que se detiene.
- (D) El objeto no se mueve al principio, después se mueve hacia atrás y finalmente se detiene.
- (E) El objeto se mueve sobre una superficie horizontal, luego se mueve hacia atrás por una pendiente y después sigue moviéndose.

Pregunta 1

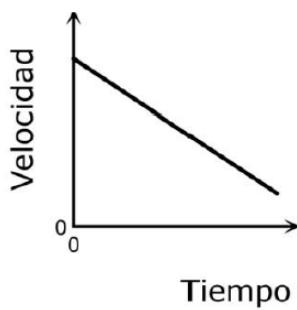
La gráfica muestra la posición en función del tiempo del movimiento de un objeto. ¿Cuál de las frases es la mejor interpretación?



- (A) El objeto se mueve con aceleración constante y diferente de cero.
- (B) El objeto no se mueve.
- (C) El objeto se mueve con una velocidad que aumenta uniformemente.
- (D) El objeto se mueve con velocidad constante.
- (E) El objeto se mueve con una aceleración que aumenta uniformemente.

Pregunta 11

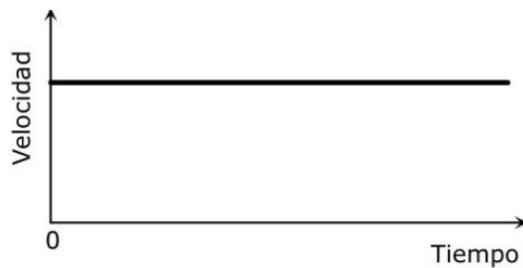
La gráfica representa la velocidad del movimiento de un objeto. ¿Cuál de las siguientes frases es la mejor interpretación?



- (A) El objeto se mueve con una aceleración constante.
- (B) El objeto se mueve con una aceleración que disminuye uniformemente.
- (C) La posición del objeto disminuye uniformemente.
- (D) El objeto se mueve a una velocidad constante.
- (E) El objeto no se mueve.

Pregunta 7

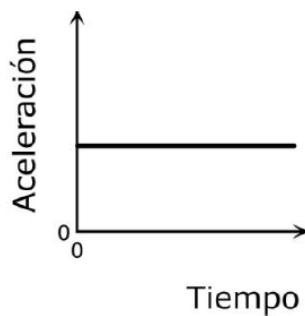
La siguiente gráfica muestra la velocidad de un objeto que se mueve en línea recta. ¿Cuál de las siguientes frases es la mejor interpretación?



- (A) El objeto se mueve incrementando su posición uniformemente.
- (B) La posición del objeto es constante.
- (C) El objeto se mueve incrementando su aceleración uniformemente.
- (D) El objeto se mueve con aceleración constante diferente de cero.
- (E) El objeto se mueve con una velocidad que aumenta uniformemente.

Pregunta 12

La gráfica representa la aceleración en función del tiempo para un objeto en movimiento. ¿Cuál de las siguientes frases es la mejor interpretación?



- (A) El objeto se mueve incrementando su aceleración uniformemente.
- (B) El objeto no se mueve.
- (C) El objeto se mueve incrementando su velocidad uniformemente.
- (D) El objeto se mueve a velocidad constante.
- (E) La posición del objeto se incrementa uniformemente.

Tabla 4.3: Resultados en describir movimiento a partir de una gráfica. La tabla contiene la cantidad de estudiantes que eligió cada opción dentro de cada pregunta, y se marca en negrita cuál es la opción correcta. También se indica la d de Cohen para cada una de las preguntas.

Pregunta 2		Pregunta 1		Pregunta 11		Pregunta 7		Pregunta 12		
Pre test	Post test	Pre test	Post test	Pre test	Post test	Pre test	Post test	Pre test	Post test	
A	20	5	13	9	5	13	12	29	11	8
B	11	16	2	1	58	40	25	27	15	9
C	8	8	15	14	11	17	2	0	11	30
D	23	48	37	46	3	7	36	19	40	28
E	16	1	11	8	1	1	3	3	1	3
<i>d</i>	0,68		0,23		0,33		0,51		0,58	

Resultados

Los resultados obtenidos por los estudiantes en esta categoría se encuentran en la Tabla 4.3.

En la pregunta 2, la opción D, que era la correcta, fue la más seleccionada por los estudiantes tanto en el pretest como en el postest. Además, se registró un aumento considerable en la proporción de respuestas correctas, con un tamaño del efecto de $d = 0,68$, lo que indica un cambio conceptual de magnitud moderada. Las opciones A y E mostraron los descensos más pronunciados. En ambas, los estudiantes interpretan erróneamente que el movimiento del objeto ocurre en dos dimensiones cuando en realidad es unidimensional. Esto refleja una disminución significativa en la tendencia a considerar la gráfica como una representación literal de la trayectoria física. La opción B fue la única incorrecta que incrementó su frecuencia de selección. Sin embargo, puede considerarse la menos inconsistente conceptualmente, ya que implica reconocer que solo hay movimiento en el segundo intervalo de tiempo, aunque se asume de manera incorrecta que el objeto “cae rodando”. Si se analizara el problema desde un enfoque dinámico, podría concluirse que esta gráfica no representa ese tipo de movimiento, pero no era esperable que los estudiantes llegaran a esa distinción, dado que cursaban su primer año de física y aún no se habían abordado los conceptos relacionados con la dinámica. En consecuencia, los estudiantes que eligieron esta opción lograron identificar los aspectos esenciales de la situación: que inicialmente no hay movimiento, que en el intervalo intermedio sí lo hay y que al final el objeto vuelve a detenerse. Por el aumento observado en la selección de esta respuesta, es razonable suponer que el cambio conceptual

alcanzado por los estudiantes fue ligeramente mayor al reflejado por el valor de d calculado.

En la pregunta 1 se obtuvo un tamaño del efecto de $d = 0,23$, el más bajo dentro de esta categoría. Si bien este valor indica un cambio de magnitud reducida, ello se debe a que en el pretest la proporción de respuestas correctas ya era alta. Este resultado puede atribuirse a que, antes de la aplicación del test, los estudiantes habían participado en una secuencia de actividades centrada en movimientos con velocidad constante. Por lo tanto, el bajo valor de d no resulta preocupante, sino coherente con el hecho de que el contenido evaluado ya había sido trabajado previamente.

Los resultados correspondientes a la pregunta 11 fueron los menos favorables del test y constituyen el único caso que puede considerarse especialmente problemático. El valor de $d = 0,33$ indica un cambio de magnitud reducida, y la cantidad de respuestas correctas en el postest fue baja (13 estudiantes). En ambas instancias, la opción más seleccionada fue la B, que refleja la interpretación de que la velocidad y la aceleración son magnitudes equivalentes o que presentan un comportamiento similar (si una disminuye uniformemente, la otra también). Este patrón de respuestas sugiere la necesidad de realizar ajustes en las actividades implementadas o de incorporar nuevas instancias que profundicen en el análisis de la aceleración a partir de gráficas de velocidad en función del tiempo.

En la pregunta 7, los resultados muestran una mejora respecto del ítem anterior. El valor de $d = 0,51$ indica un cambio conceptual de magnitud moderada. La opción D fue la más elegida en el pretest y, aunque su frecuencia disminuyó notablemente en el postest, continuó siendo una de las respuestas más seleccionadas. En esta alternativa los estudiantes vuelven a considerar que la aceleración se comporta de la misma forma que la velocidad o que ambas magnitudes son equivalentes, lo que resulta coherente con las tendencias observadas en la pregunta anterior. La opción incorrecta más elegida en el postest fue la B, que corresponde a suponer que la posición se comporta del mismo modo que la velocidad o que ambas permanecen constantes.

Por último, en la pregunta 12, el valor de $d = 0,58$ también indica un cambio conceptual de magnitud moderada. En este ítem, la opción incorrecta más frecuente fue la D, que nuevamente refleja la idea de que la velocidad y la aceleración se comportan de manera similar o que son la misma magnitud.

En conjunto, esta categoría de preguntas presentó un promedio de $\bar{d} = 0,44$,

lo que corresponde a cambios conceptuales de magnitud intermedia. Las respuestas incorrectas más frecuentes fueron aquellas que suponen que la posición, la velocidad y la aceleración se comportan de la misma forma o que son equivalentes. Este patrón coincide con errores conceptuales ampliamente documentados en la literatura, particularmente en estudios sobre interpretación de gráficas cinemáticas (McDermott et al. 1987; Zavala et al. 2019). Dichas investigaciones señalan que los estudiantes tienden a interpretar las gráficas como representaciones pictóricas del movimiento más que como descripciones funcionales, lo que explica la persistencia de esta confusión incluso después de la instrucción formal.

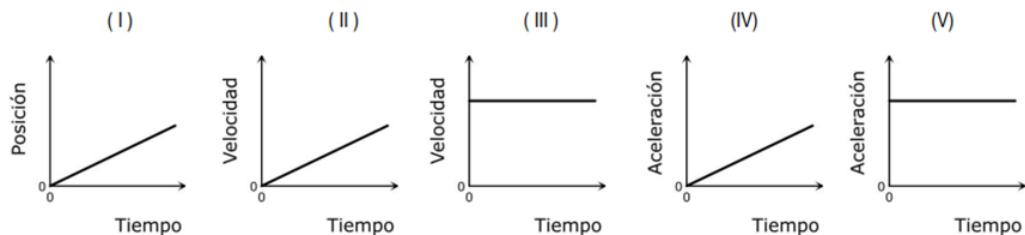
4.1.2.3. Seleccionar gráficas a partir de descripción de movimientos

En esta categoría de preguntas, los estudiantes reciben la descripción de un movimiento y deben seleccionar todas las gráficas que se correspondan con ella. Una de las preguntas describe un movimiento con velocidad constante, otra con aceleración constante, una tercera en la que la velocidad aumenta de manera uniforme y una última en la que la aceleración incrementa uniformemente.

En esta sección se presentan las cuatro preguntas junto con los resultados obtenidos por los estudiantes.

Pregunta 4

Considera las siguientes gráficas, observando las magnitudes graficadas en los diferentes ejes:



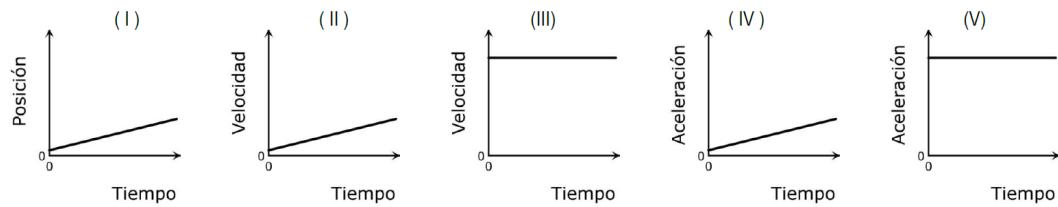
¿Cuál(es) de ellas representa(n) un movimiento a velocidad constante?

- (A) I, II y IV
- (B) I y III
- (C) Sólo III

- (D) III y V
 (E) I, III y V

Pregunta 10

Considera las siguientes gráficas, observando las magnitudes graficadas en los diferentes ejes:

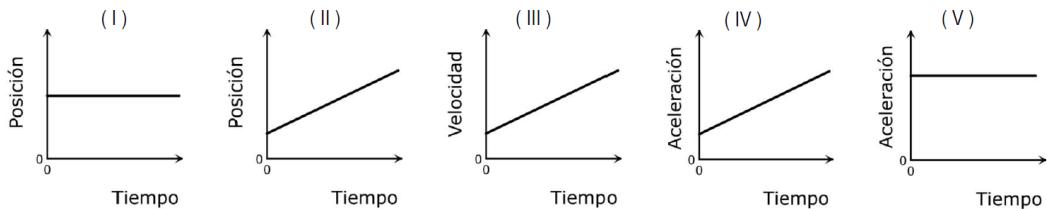


¿Cuál(es) de ellas representa(n) un movimiento a aceleración constante diferente de cero?

- (A) I, II y IV
 (B) Sólo V
 (C) II y V
 (D) Sólo IV
 (E) III y V

Pregunta 13

Considera las siguientes gráficas, observando las magnitudes graficadas en los diferentes ejes:

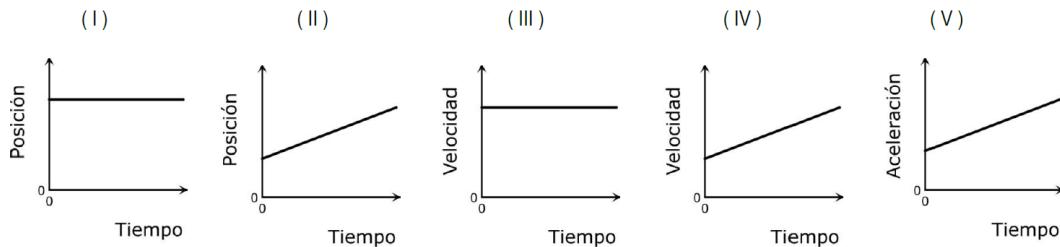


¿Cuál(es) de ellas representa(n) el movimiento de un objeto con una velocidad que se incrementa uniformemente?

- (A) Sólo II
 (B) III y V
 (C) Sólo IV
 (D) II, III y IV
 (E) Sólo III

Pregunta 8

Considera las siguientes gráficas, observando las magnitudes graficadas en los diferentes ejes:



¿Cuál(es) de ellas representa(n) un movimiento en que su aceleración se incrementa uniformemente?

- (A) II y III
- (B) IV y V
- (C) Sólo V
- (D) II, IV y V
- (E) Sólo IV

Resultados

Los resultados obtenidos por los estudiantes en esta categoría se encuentran en la Tabla 4.4.

Tabla 4.4: Resultados en seleccionar gráfica a partir de una descripción. La tabla contiene la cantidad de estudiantes que eligió cada opción dentro de cada pregunta, y se marca en negrita cuál es la opción correcta. También se indica la d de Cohen para cada una de las preguntas.

	Pregunta 4		Pregunta 10		Pregunta 13		Pregunta 8	
	Pre test	Post test	Pre test	Post test	Pre test	Post test	Pre test	Post test
A	21	15	17	10	2	2	2	2
B	6	21	9	14	7	26	15	17
C	7	12	8	32	6	7	19	27
D	34	16	9	3	42	25	39	22
E	10	14	35	19	21	18	3	10
d	0,53		0,75		0,62		0,23	

En la pregunta 4, el valor de $d = 0,53$ indica un cambio conceptual de magnitud moderada. Las respuestas del postest se distribuyeron de manera heterogénea, sin que una opción predominara claramente, lo que dificulta identificar una concepción alternativa dominante entre los estudiantes. Sin embargo, se observa un descenso importante en la cantidad de respuestas correspondientes a la opción D, que incluía las dos gráficas con una recta horizontal. Esta disminución resulta significativa, ya que sugiere que menos estudiantes consideran que la aceleración y la velocidad se comportan de la misma manera o que no distinguen entre ambas magnitudes. La opción A también refleja ese tipo de confusión, puesto que quienes la seleccionaron suponen además que la posición varía del mismo modo; esta respuesta también mostró un descenso en el postest.

Los resultados de la pregunta 10 evidencian un cambio conceptual de mayor magnitud, con un valor de $d = 0,75$. En este caso, las opciones A y E, que presentan gráficas con la misma forma para magnitudes diferentes, mostraron descensos notables en su frecuencia de selección. La única respuesta incorrecta que aumentó levemente fue la B, en la cual se selecciona una de las dos gráficas correctas, pero no ambas. Este patrón sugiere una comprensión parcial del problema, aunque conceptualmente más consistente que en otras opciones.

La tendencia observada, caracterizada por la disminución en la selección de opciones que presentan todas las gráficas con la misma forma, se mantiene en la pregunta 13. En este ítem, la opción D experimentó un descenso considerable y el valor de $d = 0,62$ corresponde a un cambio conceptual de magnitud moderada.

En la pregunta 8 también se observa un descenso importante en la selección de la opción D, que incluía todas las gráficas con la misma forma. No obstante, el aumento en la cantidad de estudiantes que eligieron la opción E resulta menos favorable. En esta alternativa, los estudiantes seleccionaron una gráfica de velocidad en función del tiempo que aumenta uniformemente para representar un movimiento en el que la aceleración también aumenta de manera uniforme, lo que indica que continúan considerando que la velocidad y la aceleración se comportan de forma similar. De hecho, también se registró un leve incremento en las respuestas correspondientes a la opción B, que incluye la misma gráfica de velocidad y otra de aceleración con igual forma, lo cual es coherente con la misma confusión. En conjunto, estos resultados sugieren que los estudiantes logran distinguir con mayor claridad la posición de las otras magnitudes, aunque

aún presentan dificultades para diferenciar entre velocidad y aceleración.

En esta pregunta, el valor de $d = 0,23$ indica un cambio de magnitud reducida. Si bien este resultado muestra una mejora limitada, debe considerarse que el ítem aborda un movimiento con aceleración variable, un tipo de situación que no se trabajó explícitamente en las actividades desarrolladas. En consecuencia, era previsible que en esta pregunta no se observaran cambios conceptuales significativos.

En esta categoría de preguntas se obtuvo un valor promedio de $\bar{d} = 0,53$, que corresponde a cambios conceptuales de magnitud moderada. En todas las preguntas se registró una disminución en la cantidad de estudiantes que seleccionaron opciones con gráficas de igual forma, lo que indica que, tras las actividades implementadas, lograron diferenciar mejor los comportamientos de la posición, la velocidad y la aceleración. Sin embargo, los resultados también sugieren que persisten mayores dificultades en la distinción entre velocidad y aceleración, aspecto que debería considerarse en futuras revisiones o ampliaciones de la secuencia didáctica.

4.1.3. Resultados globales del test

Además de analizar la proporción de respuestas correctas en cada pregunta, examinamos el número total de respuestas correctas obtenidas por cada estudiante. Estos resultados se presentan en el histograma de la Fig. 4.1.

En el pretest, los estudiantes obtuvieron puntuaciones bajas, con una media de 2,3 y un 68 % de ellos con dos respuestas correctas o menos. Estos resultados eran esperables, dado que se trataba del primer curso de física y las preguntas abordaban conceptos que aún no habían sido trabajados. Resulta llamativo que un estudiante respondió correctamente las trece preguntas en esta instancia y otro doce; ambos lograron responder correctamente todas las preguntas en el postest.

En el postest, las puntuaciones mejoraron de manera general, con una media de 5,1. Además, se observa una distribución más equilibrada y una menor concentración de resultados muy bajos. Aunque los valores siguen siendo relativamente modestos, esto resulta razonable considerando que se trata de estudiantes de 14 y 15 años en su primer curso de física. El TUGK suele aplicarse a estudiantes de niveles más avanzados, de bachillerato o universitarios, que no solo poseen mayor madurez cognitiva y experiencia en la disciplina, sino

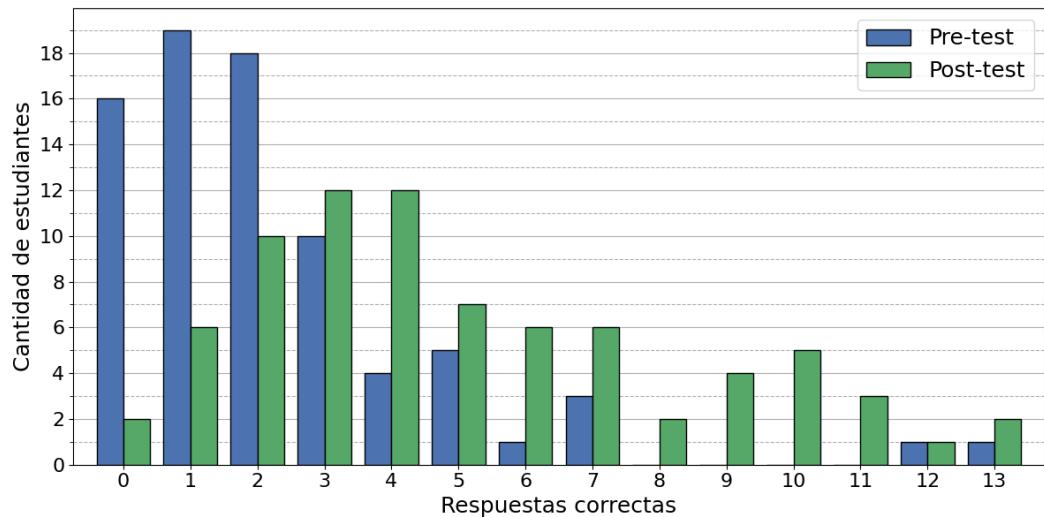


Figura 4.1: Histograma de la cantidad de respuestas correctas por estudiante. En el pre test hay una gran concentración de estudiantes con 2 respuestas correctas o menos, mientras que en el post test la distribución es más amplia y con mejores resultados. En el pre test la media fue 2,3, mientras que en el post test fue 5,1.

que además han optado voluntariamente por formaciones en las que la física ocupa un lugar central. En contraste, los estudiantes que participaron en esta investigación cursan física como asignatura obligatoria, por lo que presentan una diversidad de intereses y motivaciones hacia la materia.

4.2. Evaluación mediante rúbrica

Los aprendizajes en cinemática no fueron evaluados únicamente mediante el test. Considerando que estos contenidos forman parte del curso y que los resultados del test no incidieron en la calificación final de los estudiantes, se implementó una evaluación adicional con fines calificatorios. Esta segunda instancia permite, además, establecer una comparación entre los desempeños observados en la evaluación formal del curso y los resultados obtenidos en el test.

En esta actividad se le otorgó a los estudiantes una gráfica de posición en función del tiempo, y se les solicitó:

- Explicar cómo se debería mover la persona para replicar esta gráfica usando un sensor de posición.
- Para el movimiento correspondiente a esta gráfica, realizar un bosquejo de $v=f(t)$ y $a=f(t)$.

- Indicar qué ocurre con la posición, con la velocidad, con la aceleración, hacia dónde es el movimiento y cuál es su nombre.

Se utilizaron siete gráficas diferentes, a cada estudiante le tocaba realizar la actividad con una de ellas. Lo que tenían todas en común era que el movimiento se encontraba dividido en tres intervalos de tiempo, uno en el que el movimiento era con velocidad constante, uno con reposo y uno con aceleración constante. Una de las gráficas se encuentra en la Fig. 4.2.

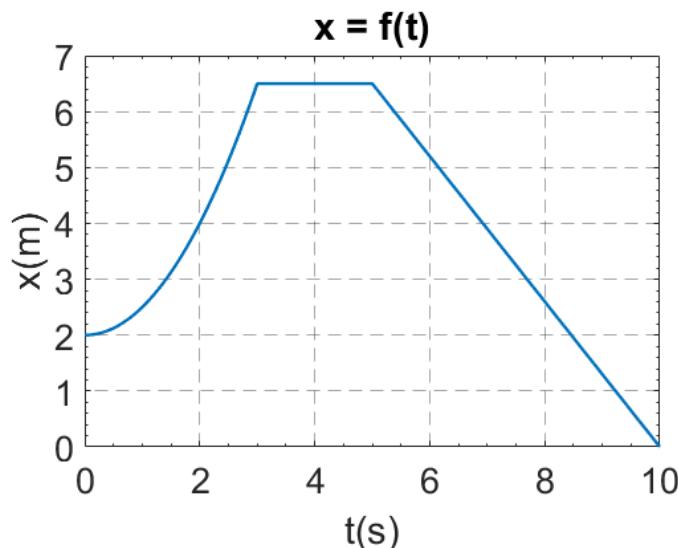


Figura 4.2: Gráfica utilizada en la evaluación. En el primer intervalo el movimiento es con aceleración constante y alejándose del sensor, en el segundo es reposo y en el tercero es con velocidad constante y acercándose al sensor.

Los estudiantes dispusieron de 40 minutos para realizar la actividad, y la hicieron en papel y sin material. Las respuestas fueron evaluadas mediante rúbricas, que se encuentran en las Tablas 4.5, 4.6 y 4.7, y los resultados obtenidos por los estudiantes en las Figuras 4.3, 4.4 y 4.5 respectivamente.

Un estudiante que obtuvo un avance destacado en las tres rúbricas realizó la siguiente descripción:

Para realizar este movimiento la persona debe comenzar cerca del sensor e ir alejándose cada vez más rápido. Luego debe quedarse quieta y por último debe acercarse al sensor con una velocidad constante.

En el primer intervalo la posición aumenta y es positiva, la velocidad va aumentando y es positiva y la aceleración es positiva y

Tabla 4.5: Rúbrica de evaluación del intervalo en reposo. Se muestra la categorización de los distintos niveles de avance.

Nivel de avance	Descripción
Mínimo	No identifica el intervalo en el que la persona se mantiene en reposo.
Escaso	Identifica el intervalo en el que la persona se mantiene en reposo, pero no menciona que la velocidad y la aceleración son nulas en este intervalo.
Moderado	Identifica el intervalo en el que la persona se mantiene en reposo. Menciona que la velocidad o la aceleración es nula y la grafica correctamente, o menciona que ambas son nulas pero no logra graficarlas.
Significativo	Identifica el intervalo en el que la persona se mantiene en reposo. Menciona que la velocidad y la aceleración son nulas, y las representa de forma parcialmente correcta en gráficas de $v = f(t)$ y $a = f(t)$.
Destacado	Identifica el intervalo en el que la persona se mantiene en reposo. Menciona que la velocidad y la aceleración son nulas, y las representa correctamente en gráficas de $v = f(t)$ y $a = f(t)$.

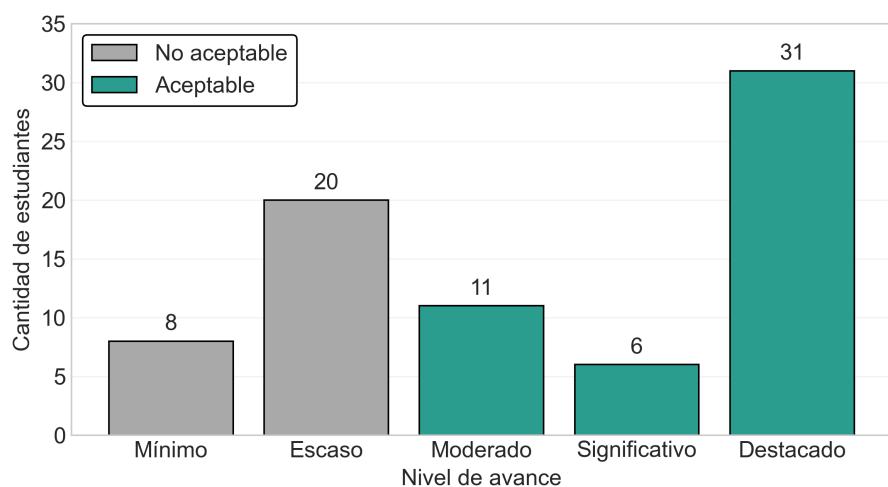


Figura 4.3: Histograma con los niveles de avance de los estudiantes en el intervalo con reposo. El 63 % de los estudiantes llegaron al menos al nivel de avance moderado.

Tabla 4.6: Rúbrica de evaluación del intervalo con movimiento rectilíneo uniforme. Se muestra la categorización de los distintos niveles de avance.

Nivel de avance	Descripción
Mínimo	No identifica qué ocurre en el intervalo en el que la persona se encuentra en un MRU.
Escaso	Identifica que en este intervalo la persona se mueve, y una de las siguientes características: el sentido del movimiento, que la velocidad es constante y que la aceleración es nula.
Moderado	Identifica que en este intervalo la persona se mueve con velocidad constante. Además, reconoce el sentido del movimiento, que la aceleración es nula, o realiza correctamente las gráficas de $v = f(t)$ y $a = f(t)$.
Significativo	Identifica que en este intervalo la persona se mueve con velocidad constante. Además, reconoce el sentido del movimiento y que la aceleración es nula, o realiza correctamente las gráficas de $v = f(t)$ y $a = f(t)$.
Destacado	Identifica que en este intervalo la persona se mueve con velocidad constante. Además, reconoce el sentido del movimiento, que la aceleración es nula y realiza correctamente las gráficas de $v = f(t)$ y de $a = f(t)$.

constante. El movimiento que hace es alejándose del sensor y es un MRUA. En el segundo intervalo la posición no cambia, no hay velocidad ni aceleración porque la persona debe quedarse quieta. El nombre del movimiento es reposo. En el tercer y último intervalo la posición disminuye, la velocidad es constante y negativa y no hay aceleración. El movimiento es hacia el sensor y se llama MRU.

Los resultados obtenidos por los estudiantes en esta actividad fueron considerablemente mejores que los registrados en el test. Si bien esta instancia no permite cuantificar con precisión los aprendizajes, ya que se realizó una sola vez, los resultados sugieren que el desempeño de los estudiantes podría reflejar una comprensión más sólida de los conceptos evaluados. Este contraste puede explicarse por diversos factores: la falta de familiaridad con el formato de preguntas de opción múltiple, una menor implicación en el test al no tener incidencia en la calificación, el hecho de que los estudiantes se prepararan específicamente para esta actividad, o incluso otros motivos que no se pudieron identificar con certeza.

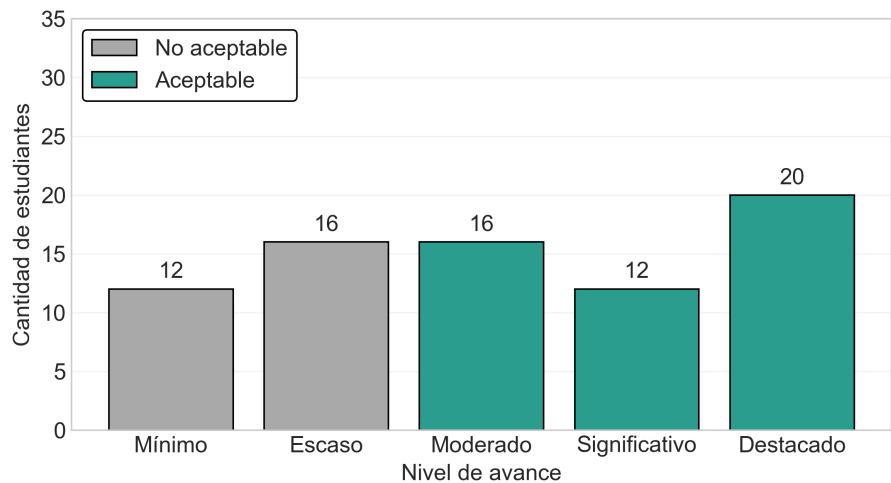


Figura 4.4: Histograma con los niveles de avance de los estudiantes en el intervalo con velocidad constante. El 63 % de los estudiantes llegaron al menos al nivel de avance moderado.

Tabla 4.7: Rúbrica de evaluación del intervalo en movimiento rectilíneo uniformemente acelerado. Se muestra la categorización de los distintos niveles de avance.

Nivel de avance	Descripción
Mínimo	No identifica qué ocurre en el intervalo en el que la persona se encuentra en un MRUA.
Escaso	Identifica que en este intervalo la persona se mueve, y una de las siguientes características: el sentido del movimiento, que la velocidad varía uniformemente y que la aceleración es constante.
Moderado	Identifica que en este intervalo la persona se mueve con aceleración constante. Además, reconoce el sentido del movimiento, que la velocidad varía uniformemente, o realiza correctamente las gráficas de $v = f(t)$ y $a = f(t)$.
Significativo	Identifica que en este intervalo la persona se mueve con aceleración constante. Además, reconoce el sentido del movimiento y que la velocidad varía uniformemente, o realiza correctamente las gráficas de $v = f(t)$ y $a = f(t)$.
Destacado	Identifica que en este intervalo la persona se mueve con aceleración constante. Además, reconoce el sentido del movimiento, que la velocidad varía uniformemente y realiza correctamente las gráficas de $v = f(t)$ y $a = f(t)$.

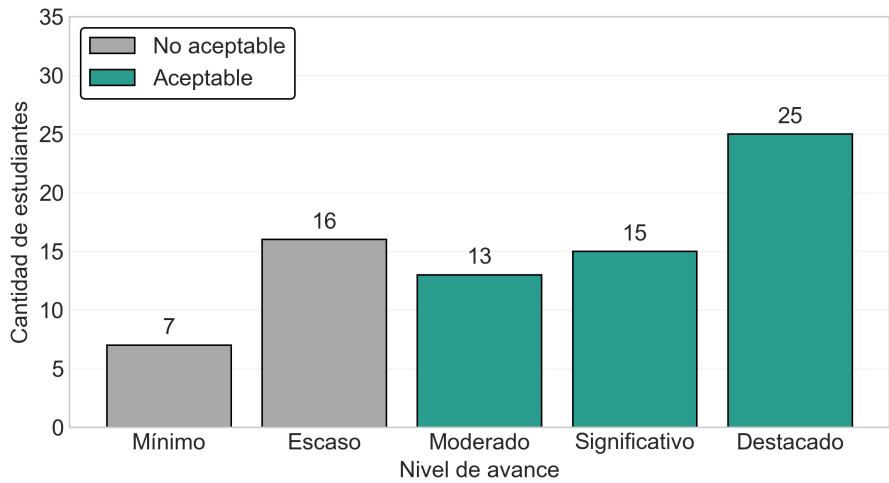


Figura 4.5: Histograma con los niveles de avance de los estudiantes en el intervalo con aceleración constante. El 70 % de los estudiantes llegaron al menos al nivel de avance moderado.

4.3. Discusión de los resultados

La evidencia combinada del test y de la evaluación con rúbrica indica cambios conceptuales de magnitud intermedia y un perfil de aprendizaje heterogéneo. En el test, los tamaños del efecto promedio por categoría fueron $d = 0,56$ para la selección de una gráfica a partir de otra, $d = 0,44$ para la descripción de un movimiento a partir de una gráfica y $d = 0,53$ para la selección de gráficas a partir de descripciones, con un valor global cercano a $d = 0,50$. El número promedio de respuestas correctas por estudiante aumentó de 2,3 a 5,1 sobre un total de 13 ítems, acompañado de una menor concentración de puntajes en los niveles más bajos.

En la evaluación mediante las rúbricas, al menos el 63 % alcanzó nivel moderado o superior en reposo y MRU, y el 70 % en MRUA, lo que sugiere que, ante tareas de producción y justificación, los estudiantes movilizan mejor los vínculos entre representaciones que en formato de opción múltiple.

En conjunto, se consolidó el uso de la pendiente para conectar $x(t)$ con $v(t)$ y la distinción entre posición y velocidad, mientras que persisten dificultades al articular velocidad y aceleración, al comparar magnitudes relativas y en situaciones con aceleración variable. Esto orienta ajustes didácticos concretos: incluir más actividades de comparación entre $v(t)$ y $a(t)$, trabajar explícitamente la escala además de la forma de las gráficas e introducir gradualmente casos con $a(t)$ variable.

Finalmente, las diferencias de formato y consecuencias evaluativas ayudan a explicar la brecha entre ambos instrumentos: el test capta con precisión concepciones alternativas persistentes, mientras que la rúbrica revela competencias que el test tiende a subestimar.

Capítulo 5

Evaluación del desarrollo del PC

Este capítulo analiza el desarrollo de habilidades de PC en el estudio del movimiento acelerado mediante dos contextos complementarios: una actividad sobre MRUA que propone una aproximación iterativa por intervalos (construcción de una tabla y, luego, explicación del programa que automatiza ese procedimiento) y una actividad de caída libre centrada en la interpretación de un programa en un contexto físico distinto. El objetivo es caracterizar la progresión de los estudiantes y las dificultades recurrentes al traducir la física al código, construir algoritmos iterativos, generar datos, aplicar la lógica computacional, y vincular representaciones tabulares con representaciones computacionales. Para ello se emplean rúbricas alineadas con el marco de Weller et al. (2022), junto con un análisis mixto: resultados cuantitativos (histogramas de niveles de logro) y evidencia cualitativa (fragmentos de respuestas). Cabe señalar que los grupos participantes difieren entre contextos, por lo que las comparaciones se interpretan como contrastes complementarios más que como seguimiento a través de un mismo curso.

5.1. Evaluación sobre algoritmos de MRUA en Python

Dentro de la secuencia de actividades presentada en el capítulo 3, una de ellas se centró en el desarrollo de un programa en Python que genera listas de tiempos, velocidades y posiciones de un objeto que se mueve con aceleración constante, es decir, en un MRUA. Dado que dicha actividad se realizó en grupos, se propuso posteriormente una tarea individual con un propósito similar,

destinada a evaluar la comprensión del programa elaborado colectivamente. En esta nueva instancia, los estudiantes debían completar una tabla de tiempo, velocidad y posición correspondiente a un MRUA, calculando los valores manualmente, y luego explicar el funcionamiento de un programa que realiza el mismo procedimiento.

A los efectos de analizar el desarrollo de PC, esta sección se organiza en dos partes complementarias: la elaboración manual de una tabla mediante un procedimiento iterativo por intervalos, y la explicación del programa en Python que automatiza ese mismo proceso. En ambos casos se emplean rúbricas específicas y se reportan resultados cuantitativos y cualitativos.

5.1.1. Primera parte: Elaboración de la tabla

La consigna fue la siguiente:

Una persona se mueve en línea recta y con aceleración constante. Su posición inicial es $x_i = 10$ m, su velocidad inicial es $v_i = 3$ m/s y su aceleración es $a = 4$ m/s². Realizando la misma aproximación que usamos en clase (suponer que la velocidad se mantiene constante durante 0,1 s y que luego cambia de golpe), completa la siguiente tabla:

t (s)	v (m/s)	x (m)
0		
0,1		
0,2		

Debes incluir todos los cálculos que realizaste para hallar cada valor.

Las respuestas de los estudiantes fueron evaluadas mediante una rúbrica, presentada en la Tabla 5.1, y los resultados obtenidos se muestran en la Figura 5.1.

En el nivel mínimo de la rúbrica correspondiente a la elaboración de la tabla, los estudiantes no lograron completar adecuadamente la tarea. Por ejemplo, un estudiante presentó la siguiente tabla:

t (s)	v (m/s)	x (m)
0	1	10
0.1	2	20
0.2	3	30

En este caso, el único valor correctamente identificado es la posición inicial. El resto de los valores son incorrectos y, además, el estudiante no incluye ninguna justificación o explicación del procedimiento seguido, lo que impide inferir su razonamiento.

En el nivel escaso se observa un mayor grado de avance, ya que los estudiantes lograron completar parcialmente la tabla de forma correcta. Por ejemplo, un estudiante elaboró la siguiente tabla:

t (s)	v (m/s)	x (m)
0	3	10
0, 1	$4, 9 = 3 + 4^2 \cdot 0, 1$	$12, 6 = 10 + 4^2 \cdot 0, 1$
0, 2	$8, 18 = 4, 9 + 4^2 \cdot 0, 2$	$17, 72 = 12, 6 + 4^2 \cdot 0, 2$

Este estudiante identificó correctamente los valores iniciales y el modo de calcular la velocidad en la segunda fila. Sin embargo, interpretó erróneamente que la aceleración era 4^2 en lugar de 4, probablemente por considerar que el exponente de la unidad (m/s^2) debía aplicarse también al valor numérico. Al calcular la posición, utilizó el valor de la aceleración en lugar de la velocidad anterior. Sus planteos no reflejan los resultados obtenidos: en la primera velocidad, el error parece deberse a un orden incorrecto en las operaciones, posiblemente al ingresar los valores en la calculadora como $(3+4)^2 \cdot 0,1$, mientras que en las demás celdas no resulta claro el motivo de la discrepancia entre el planteo y el resultado. En la última fila repite los mismos errores y, además, emplea 0,2 s en lugar de 0,1 s, lo que sugiere una confusión entre el instante y el intervalo de tiempo.

En el nivel moderado los estudiantes lograron realizar tablas parcialmente correctas. Por ejemplo, uno de ellos presentó la siguiente resolución:

t (s)	v (m/s)	x (m)
0	3	10
0, 1	3	$10, 3 = 10 + 3 \cdot 0, 1$
0, 2	$7 = 3 + 4$	$11 = 10, 3 + 7 \cdot 0, 1$

En este caso, el estudiante identificó correctamente los valores iniciales y el procedimiento para calcular las posiciones. Sin embargo, los valores de velocidad no fueron obtenidos de manera adecuada, lo que condujo a un error en la última posición, ya que esta depende directamente de la velocidad de la fila anterior.

En el nivel significativo los estudiantes lograron realizar tablas mayormente correctas. Por ejemplo, uno de ellos presentó la siguiente resolución:

t (s)	v (m/s)	x (m)
0	3	10
0, 1	$3, 4 = 3 + 4 \cdot 0, 1$	$10, 3 = 10 + 3 \cdot 0, 1$
0, 2	$4, 2 = 3, 4 + 4 \cdot 0, 2$	$10, 98 = 10, 3 + 3, 4 \cdot 0, 2$

En este caso, el estudiante aplicó correctamente el procedimiento de cálculo y comprendió la relación entre las variaciones de velocidad y posición en intervalos sucesivos. El único error identificado se encuentra en la última fila, donde utiliza 0, 2 s en lugar de 0, 1 s para el incremento temporal, lo que evidencia una confusión entre el instante y el intervalo de tiempo considerado.

En el nivel destacado los estudiantes realizaron la tabla de forma completamente correcta. Un ejemplo representativo se muestra a continuación:

t (s)	v (m/s)	x (m)
0	3	10
0, 1	$3, 4 = 3 + 4 \cdot 0, 1$	$10, 3 = 10 + 3 \cdot 0, 1$
0, 2	$3, 8 = 3, 4 + 4 \cdot 0, 1$	$10, 64 = 10, 3 + 3, 4 \cdot 0, 1$

En este caso, el estudiante identificó correctamente las condiciones iniciales y aplicó sin errores el procedimiento iterativo para calcular las velocidades y posiciones en cada intervalo de tiempo. Los cálculos fueron presentados de manera ordenada y coherente, mostrando una comprensión sólida tanto de la relación entre las magnitudes físicas como del sentido del método empleado para aproximar el movimiento acelerado mediante pasos discretos.

Tabla 5.1: Rúbrica de evaluación: elaboración de la tabla.

Nivel de avance	Descripción
Mínimo	No logra realizar la tabla.
Escaso	Realiza de forma parcialmente correcta alguna parte de una tabla de velocidad y posición en función del tiempo de un movimiento rectilíneo uniformemente acelerado a partir de sus condiciones iniciales.
Moderado	Realiza de forma parcialmente correcta una tabla de velocidad y posición en función del tiempo de un movimiento rectilíneo uniformemente acelerado a partir de sus condiciones iniciales.
Significativo	Realiza correctamente la tabla de velocidad y posición en función del tiempo, pero comete un error menor en alguno de los planteos o en los valores obtenidos.
Destacado	Realiza correctamente una tabla de velocidad y posición en función del tiempo de un movimiento rectilíneo uniformemente acelerado a partir de sus condiciones iniciales, incluyendo el planteo correspondiente para todos los cálculos.

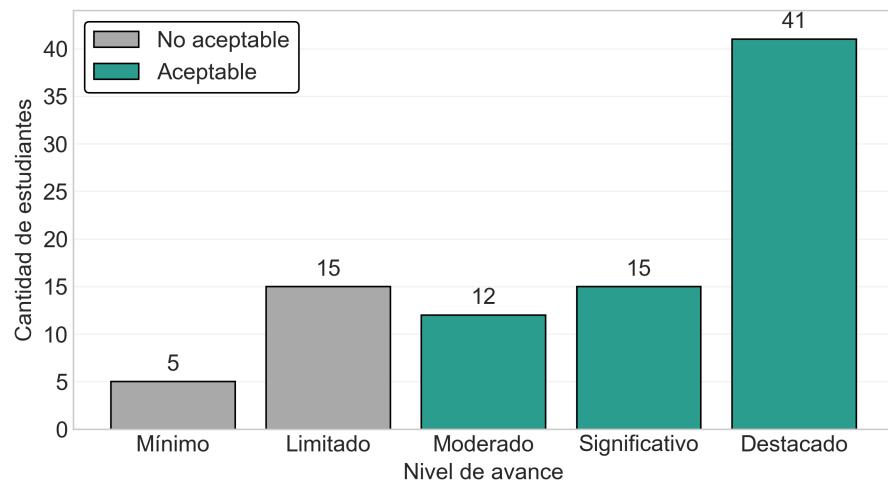


Figura 5.1: Histograma con los niveles de avance de los estudiantes en la elaboración de la tabla de tiempo, velocidad y posición de un MRUA. El 77 % de los estudiantes llegaron al menos al nivel de avance moderado.

5.1.2. Segunda parte: Explicación del programa

El análisis de las tablas elaboradas por los estudiantes permite apreciar distintos grados de comprensión del procedimiento iterativo utilizado para

aproximar el movimiento acelerado. Sin embargo, para obtener una visión del desarrollo de sus habilidades de PC, resulta necesario considerar las explicaciones que dieron sobre el programa en Python que implementa ese mismo proceso. Mientras que la elaboración de la tabla pone en juego principalmente la aplicación manual del algoritmo paso a paso, lo cual es necesario pero no suficiente, la descripción del programa requiere un nivel adicional de abstracción, ya que implica reconocer cómo el código automatiza ese procedimiento y cómo las variables y estructuras del lenguaje representan las magnitudes físicas involucradas. En síntesis, la segunda parte indaga si los estudiantes logran reconocer, en el código, la misma estructura de actualización paso a paso que aplicaron en la tabla.

En la segunda parte de la actividad, se solicitó: *Explica qué ocurre en cada una de las líneas de código cuando se ejecuta el siguiente programa.*

```

1 t_i = 0
2 t_f = 1
3 dt = 0.1
4 x_i = 10
5 v_i = 3
6 a = 4
7
8 t = [t_i]
9 v = [v_i]
10 x = [x_i]
11
12 while t[-1] < t_f:
13     t.append(t[-1] + dt)
14     v.append(v[-1] + a*dt)
15     x.append(x[-1] + v[-1]*dt)

```

Las respuestas de los estudiantes fueron evaluadas mediante una rúbrica, presentada en la Tabla 5.2, y los resultados obtenidos se muestran en la Figura 5.2.

En el nivel mínimo, las respuestas mostraron una comprensión muy limitada del funcionamiento del programa. Los estudiantes se centraron en mencionar algunos elementos de la consigna sin establecer vínculos claros con las líneas de código. Por ejemplo, un estudiante escribió:

La consigna dice que cada 0,1 s su aceleración es de 3 m/s entonces a cada resultado le tenés que sumar 3m/s por cada vez que avanza.

Lo que ocurre en las líneas de código es que se restan por -1, por t , dt , x .

Esta respuesta no logra identificar ninguna de las operaciones que realiza el código ni el rol de las estructuras utilizadas, y refleja una interpretación superficial basada únicamente en la descripción verbal del fenómeno.

En el nivel escaso, los estudiantes reconocieron algunos componentes del código, como las variables iniciales y la creación de listas, pero sin comprender el propósito del bucle ni el proceso iterativo. Un ejemplo representativo es esta parte de una respuesta: “*El while hace que se repita infinitamente la tabla, lo demás hace las sumatorias.*” Si bien el estudiante asocia el *while* con una repetición, interpreta que esta ocurre de forma infinita y no logra explicar la condición de detención ni el modo en que se actualizan las listas, lo que evidencia una comprensión parcial del funcionamiento del programa.

En el nivel moderado, los estudiantes lograron reconocer la función de las distintas líneas y relacionarlas con las magnitudes físicas involucradas. Por ejemplo, un estudiante explicó:

La línea 8 nos indica que $ti = t_i$, la 9 que $vi = v_i$ y la 10 que $xi = x_i$. En la 12 que mientras $t[-1]$ sea menor que t_f : la línea 13 está haciendo la fórmula de tiempo ($t_{anterior} + dt$), la 14 hace la fórmula de velocidad ($v_{anterior} + a * dt$) y la 15 la fórmula de posición ($x_{anterior} + v_{anterior} * dt$).

Esta respuesta muestra que el estudiante comprendió correctamente los cálculos realizados en cada paso, aunque no reconoció que se trata de listas a las que se van agregando valores, sino que las interpretó como simples asignaciones de variables.

En el nivel significativo, los estudiantes ofrecieron explicaciones más detalladas, identificando adecuadamente las variables iniciales, las listas y el papel general del bucle. Un ejemplo de este nivel es la respuesta:

Las líneas 1, 2, 3, 4, 5 y 6 sirven para anotar los datos iniciales. Dice que va de 0 a 1 segundos con una posición inicial de 10 metros y una velocidad inicial de 3, también con una aceleración de 4 y $dt=0.1$, o sea que cada 0.1 segundos de 0 a 1 segundos se realizan los cálculos. Las líneas 8, 9 y 10 son listas para tiempo, velocidad y posición. La 12 dice que mientras el t_f sea mayor a $t[-1]$ se van

a hacer los cálculos de las líneas 13, 14 y 15. Las tres líneas finales son para realizar los cálculos de cada lista y actualizar los valores.

Esta explicación es casi completa, aunque el estudiante no menciona qué significa $t[-1]$ ni cómo se calculan los nuevos valores en cada iteración. En otro caso, un estudiante explicó correctamente el código hasta la última línea, donde escribió: “*Agresa a la lista x el resultado de: el último valor en x más lo que da v por -2 por dt.*” En este caso, interpreta erróneamente el término $v[-2]$, confundiéndolo con “ $v por -2$ ”, lo que indica una comprensión incompleta de la notación de índices en listas.

Por último, en el nivel destacado los estudiantes ofrecieron explicaciones precisas y completas, reconociendo la estructura general del programa y el propósito de cada línea de código. Un ejemplo de este nivel es la siguiente descripción:

Líneas 1-6: Se declaran las distintas variables mencionadas. Líneas

8-10: Se añaden las variables a las listas creadas. En este caso se añade t_i a t , v_i a v y x_i a x . Líneas 12-15: Se dicta que mientras el último valor sea menor que el tiempo final se repita el siguiente bucle:

- Añadir a la lista t el resultado de: último valor de t + intervalo de t .
- Añadir a la lista v el resultado de: último valor de v + aceleración*intervalo de tiempo.
- Añadir a la lista de x el resultado de: último valor de x + penúltimo valor de v *intervalo de tiempo.

En este caso, el estudiante no solo identifica correctamente la función de cada parte del código, sino que también explica de manera clara cómo se actualizan las listas y cómo el bucle implementa la aproximación iterativa del movimiento acelerado.

5.1.3. Resultados

Los resultados obtenidos en las Figuras 5.1 y 5.2 muestran que la mayoría de los estudiantes alcanzó niveles de avance significativos o destacados tanto en la elaboración de la tabla como en la descripción del programa. En la primera parte, la mayor parte logró construir correctamente la tabla de tiempos,

Tabla 5.2: Rúbrica de evaluación: descripción del programa.

Nivel de avance	Descripción
Mínimo	No logra identificar qué realiza el programa.
Escaso	Reconoce algunos elementos del código, pero no logra explicar qué ocurre al ejecutarlos.
Moderado	Explica qué ocurre en las partes más importantes del programa, aunque de forma incompleta o con errores.
Significativo	Explica la mayoría de las líneas del programa indicando qué ocurre cuando se ejecutan, aunque con algunas imprecisiones o faltantes.
Destacado	Reconoce con claridad qué ocurre cuando se ejecuta cada línea del programa: identifica la declaración de variables con las condiciones iniciales, la creación de listas que representan las columnas de la tabla y un bucle <i>while</i> en el que se calculan y agregan los valores correspondientes.

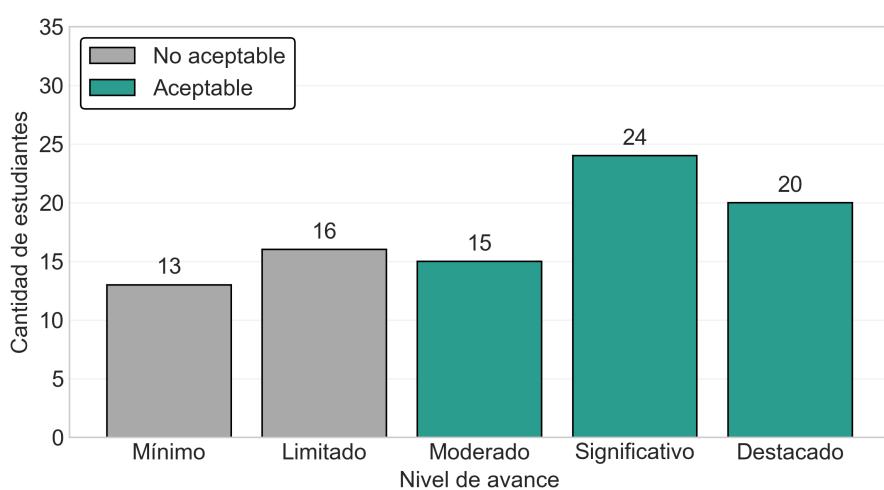


Figura 5.2: Histograma con los niveles de avance de los estudiantes en la explicación del programa en Python que elabora listas de tiempo, velocidad y posición de un MRUA. El 67 % de los estudiantes llegaron al menos al nivel de avance moderado.

velocidades y posiciones, aplicando la idea de dividir el movimiento en intervalos cortos de tiempo en los cuales la velocidad se considera constante. Esto indica que comprendieron el procedimiento iterativo utilizado para aproximar el movimiento acelerado, reconociendo la relación entre los cambios sucesivos de velocidad y posición.

En la segunda parte, correspondiente a la explicación del programa, también se observó un buen desempeño general, aunque con una mayor variabilidad entre los niveles de avance. Si bien muchos estudiantes identificaron correctamente el rol de las variables, las listas y el bucle *while*, algunos mostraron dificultades para explicar cómo el programa actualiza los valores de las magnitudes físicas en cada iteración. Estas dificultades, observadas principalmente en los niveles intermedios de la rúbrica, reflejan una comprensión parcial del funcionamiento del algoritmo y de la forma en que el código implementa la aproximación discreta del movimiento.

El desempeño alcanzado sugiere que los estudiantes pudieron establecer vínculos entre las dos representaciones del fenómeno físico, integrando habilidades de PC como la descomposición, la traducción entre la física y el código y la construcción de algoritmos. No obstante, la menor proporción de respuestas destacadas en la explicación del programa, en comparación con la elaboración de la tabla, evidencia que aún resulta desafiante comprender de manera completa la lógica del bucle y el proceso de actualización de las variables.

En conjunto, los resultados sugieren que esta actividad permitió a los estudiantes aproximarse a una forma distinta de representar el movimiento, basada en la descomposición del fenómeno en pasos discretos y en la construcción de un modelo iterativo. Si bien no todos lograron reconocer plenamente el potencial del código como herramienta para automatizar cálculos y explorar el comportamiento del sistema, varios estudiantes mostraron indicios de esta comprensión al reflexionar sobre la cantidad de operaciones que el programa realiza de manera rápida y sistemática. En este sentido, la propuesta constituyó una instancia valiosa para introducir una perspectiva computacional del movimiento y promover una comprensión más dinámica de los procesos físicos.

5.2. Explicación de programa sobre caída libre en Python

En esta sección se presenta una secuencia de actividades diseñada para evaluar el desarrollo del PC en el contexto del estudio de la caída libre. El objetivo principal fue analizar cómo los estudiantes aplican habilidades de PC al interpretar y explicar un programa en Python que modela un movimiento con aceleración constante, en un contexto diferente al abordado en la sección anterior. Los resultados de este trabajo fueron publicados en Dutra et al. (2025), artículo aceptado en *The Physics Teacher*.

Las actividades se realizaron luego de todas las descritas en el capítulo 3. No obstante, el grupo de estudiantes que participó en esta instancia no fue el mismo que en las evaluaciones anteriores (capítulo 4 y sección previa).

La secuencia constó de una primera etapa de trabajo grupal, en equipos de cuatro estudiantes, en la que desarrollaron un programa para analizar el movimiento de un objeto en caída libre. Cada parte de la actividad se vinculó con al menos una de las prácticas de PC propuestas por Weller et al. (2022). Finalmente, se llevó a cabo una evaluación individual en la que los estudiantes debieron explicar el funcionamiento de un programa que calculaba la posición del objeto en distintos instantes de tiempo durante su caída.

5.2.1. Primera etapa

La primera etapa se desarrolló en dos clases de dos horas. Previamente, se había analizado la dinámica de un objeto en caída libre, y se dedujo que es un movimiento con aceleración constante $a = g$. También se habían analizado las expresiones analíticas que permiten calcular la posición y la velocidad de un objeto en función del tiempo cuando el movimiento es un MRUA.

En esta actividad los estudiantes debieron, en primer lugar, definir una función en Python que recibiera como entradas la altura y velocidad iniciales de un objeto en caída libre y un instante de tiempo, y que devuelva su posición y velocidad en ese instante a partir de las Ecs. 5.1 y 5.2.

$$x(t) = x_i + v_i t + \frac{at^2}{2} \quad (5.1)$$

$$v(t) = v_i + at \quad (5.2)$$

Esta parte de la actividad requería traducir conceptos físicos a código.

Luego, los estudiantes debían probar la función creada, imprimiendo los resultados obtenidos para distintos valores de entrada. En esta instancia, la depuración adquirió un papel importante, ya que permitió identificar y corregir errores en el código.

En una tercera parte, se les solicitó simular un caso específico: la caída de un objeto soltado desde 3.8 m con velocidad inicial nula. Para ello, debían generar listas con los valores de tiempo, posición y velocidad, considerando intervalos de $dt = 0,05$ s entre 0 y 1 s. Esta tarea implicaba descomponer el problema en pasos más pequeños, construir algoritmos, aplicar lógica condicional y generar datos.

Finalmente, debían utilizar las listas obtenidas para representar gráficamente $x = f(t)$ y $v = f(t)$. La elaboración de las gráficas requería la habilidad de PC relacionada con la elección de formas adecuadas de representación de datos.

5.2.2. Evaluación

En la instancia de evaluación individual, cada estudiante debía analizar un programa en Python que simulaba la caída libre de un objeto desde una altura de 100 m, con velocidad inicial nula. El programa definía una función que recibía como entrada un tiempo y devolvía la posición correspondiente del objeto. Luego, mediante un bucle *while*, se calculaban las posiciones entre 0 y 5 s, en pasos de 1 s, imprimiendo los resultados en la consola.

La consigna para los estudiantes fue: *Tu tarea es examinar el código y determinar su función y propósito. Para ello debes describir qué hace y cómo está relacionado con el concepto de caída libre.*

```

1 def caida_libre(tiempo):
2     posicion_inicial = 100
3     velocidad_inicial = 0
4     aceleracion = -9.8
5     posicion = posicion_inicial + velocidad_inicial*tiempo
6         + aceleracion*(tiempo**2)/2
7     return posicion
8
9 tiempo_actual = 0
10 tiempo_final = 5
11 while tiempo_actual < tiempo_final:
12     posicion = caida_libre(tiempo_actual)
13     print("En el tiempo", tiempo_actual, "s, la posición es",
14         ", posicion, "m")
15     tiempo_actual += 1

```

Para analizar las respuestas de los estudiantes se elaboró una rúbrica específica, basada en el marco de prácticas de PC propuesto por Weller et al. (2022). La rúbrica contempló cuatro dimensiones principales: traducir física a código, construcción de algoritmos, generación de datos y aplicación de lógica condicional. Cada una de estas dimensiones fue evaluada en cinco niveles de logro, desde mínimo hasta destacado, de acuerdo con la precisión y la profundidad conceptual evidenciada en las explicaciones de los estudiantes. La descripción de los niveles de la rúbrica se encuentran en la Tabla 5.3 y los resultados obtenidos por los estudiantes en la Fig. 5.3.

En el nivel mínimo, las descripciones de los estudiantes estuvieron mayormente desconectadas del funcionamiento del programa y no abordaron habilidades fundamentales del PC. Por ejemplo, un estudiante escribió:

Lo que hace este código es como si estuviera corriendo o moviéndose una persona y lo que hace este código es ir subiendo la velocidad. Como podemos ver dice posición y empieza a sumar posición inicial y otras cosas. Abajo dice que la posición es 1, entonces lo que hizo fue sumar la velocidad, la posición inicial y la aceleración del tiempo para descubrir el tiempo actual.

En el programa no aparece ninguna “persona corriendo o moviéndose”, ni se establece la posición en 1. Otro estudiante afirmó: “*Creamos una función que sirve para calcular el tiempo de caída de un objeto.*” Estas respuestas reflejan una falta de reconocimiento sobre cómo deben demostrarse las competencias de PC, como la traducción de la física al código y la estructuración de algoritmos.

Tabla 5.3: Rúbrica de evaluación para la explicación del programa de caída libre.

Nivel de avance	Descripción
Destacado	Proporciona explicaciones precisas y completas que demuestran un dominio profundo: traduce claramente el modelo físico al código, detalla el algoritmo iterativo (construcción del algoritmo y generación de datos) y la lógica condicional utilizada en la simulación.
Significativo	Muestra una comprensión muy buena con explicaciones claras de cómo la física se traduce en código, reconociendo el algoritmo iterativo y la lógica condicional, a pesar de presentar inexactitudes menores.
Moderado	Evidencia una comprensión sólida pero incompleta al identificar la conversión de ecuaciones físicas en código e inferir la existencia de un algoritmo iterativo, con solo una mención implícita de la lógica condicional.
Escaso	Ofrece explicaciones superficiales que abordan parcialmente los elementos del pensamiento computacional, brindando descripciones vagas del algoritmo y de los procesos de generación de datos.
Mínimo	Presenta explicaciones desconectadas del funcionamiento real, sin reconocer cómo el modelo físico se traduce en código ni cómo opera el algoritmo subyacente.

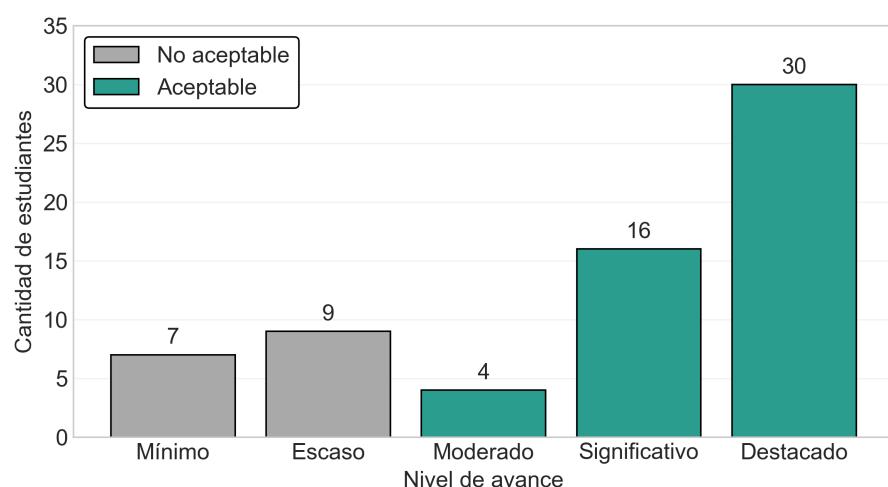


Figura 5.3: Histograma con los niveles de avance de los estudiantes en la explicación del programa. El 76 % de los estudiantes llegaron al menos al nivel de avance moderado.

En el nivel escaso, los estudiantes ofrecieron explicaciones vagas o parcialmente incorrectas que aludían a elementos del PC, pero sin desarrollarlos adecuadamente. Por ejemplo, un estudiante describió el programa así:

El propósito de este código es el calcular la caída libre que sufre un objeto cuando se suelta de determinada altura, velocidad, etc. En esta se realizan todos los procedimientos para calcular esto, para empezar se define la variable “caída_libre” como tiempo, luego se define la “posición_inicial”, la “velocidad_inicial” y la “aceleración”, después se le explica al python cómo realizar una operación para llegar al siguiente valor de posición.

Aunque esta respuesta muestra que el estudiante reconoció la necesidad de calcular valores de posición, no explica claramente el algoritmo iterativo ni la lógica condicional —competencias esenciales del PC—, y no logra articular completamente cómo la física se traduce en código.

En el nivel moderado, los estudiantes demostraron una comprensión sólida pero incompleta de los principios del PC. Por ejemplo, un estudiante explicó:

Este código sirve para saber la posición en cada instante de tiempo. Comenzando con la primera parte del código primero dice def caída libre y abajo dice los valores de la posición inicial de la velocidad inicial y de la aceleración. El def caída libre sirve para guardar todo lo que hay debajo, y luego explica la fórmula para saber la posición en cada instante de tiempo. Luego le da un valor diferente a tiempo actual y tiempo final, y muestra que posición es igual a caída libre tiempo actual. Por último, pone el print para explicar que quiere poner en el run y ahí termina.

Esta explicación identifica correctamente la conversión de una ecuación física en código e infiere el uso de un algoritmo iterativo. Sin embargo, el tratamiento implícito de la lógica condicional y cierta ambigüedad terminológica indican una comprensión parcial del conjunto de competencias del PC.

En el nivel significativo, los estudiantes demostraron una muy buena comprensión de los conceptos del PC, aunque con pequeñas imprecisiones. Un estudiante expresó:

La función de este código es determinar la posición de un objeto en cada instante de tiempo en el transcurso de una caída libre,

es decir, el programa está diseñado para que mediante diferentes cálculos que le damos nos devuelva la $x(t)$ (posición en función de tiempo) que se produzca entre los segundos 0 y 5 al producirse una caída libre en el objeto esto lo hace mediante la variable “return”. En el programa se crea una variable llamada “caida.libre” y se le brindan todos los datos necesarios para hallar este valor tras una función en la que se plantea la cuenta para calcular la posición en cada instante.

Esta respuesta muestra con claridad cómo el modelo físico se traduce en una función computacional y reconoce el papel del algoritmo iterativo y la lógica condicional. Sin embargo, una ligera confusión —como referirse a una función como si fuera una variable— resta precisión al dominio de las competencias del PC.

En el nivel destacado, los estudiantes ofrecieron explicaciones excelentes y detalladas que abordan plenamente las competencias del PC. Por ejemplo, un estudiante escribió:

En síntesis, este código lo que hace es crear una función para calcular en qué posición se encontrará un objeto en caída libre en determinado instante de tiempo y luego, para cada instante de tiempo (de 1 segundo en 1 segundo en este caso), imprimir dicho instante de tiempo y la posición (la cual es calculada mediante la función mencionada anteriormente). Lo que relaciona a este código con la caída libre es que dentro de la misma la aceleración es de $9,8 \text{ m/s}^2$. Si quisieramos programar una función para calcular la posición en cada instante de tiempo fuera de una caída libre (es decir en otro MRUA), simplemente se tendría que modificar el valor de la variable de aceleración definida dentro de la función.

Esta descripción representa con precisión cómo se implementa la ecuación de caída libre en el código, detallando el algoritmo iterativo y la lógica condicional, y vinculando claramente los datos generados con el fenómeno físico, demostrando un dominio completo de las competencias de PC.

En general, más del 75 % de los estudiantes alcanzó al menos un nivel moderado de desempeño, y el 45 % llegó al nivel destacado. Estos resultados sugieren que la mayoría de los estudiantes logró involucrarse de manera efectiva con el enfoque computacional para modelar el movimiento acelerado.

5.2.3. Conclusiones de esta actividad

Los resultados de esta actividad sugieren que los estudiantes lograron aplicar habilidades de PC para analizar el movimiento acelerado. En particular, pudieron generar correctamente las gráficas de posición en función del tiempo y de velocidad en función del tiempo para situaciones de caída libre, así como calcular el tiempo de caída de un objeto a partir de su posición y velocidad iniciales. Esto implicó poner en práctica procesos de descomposición, traducción de la física al código, construcción de algoritmos, aplicación de lógica condicional, generación y análisis de datos, depuración y trabajo colaborativo en la elaboración de modelos computacionales.

Los resultados obtenidos por los estudiantes en la evaluación individual destacan su capacidad para aplicar habilidades de PC al análisis del movimiento acelerado. Con más del 75 % de los participantes alcanzando un nivel suficiente (moderado o superior) y un 45 % llegando al nivel más alto, los hallazgos demuestran su comprensión del cálculo de la posición en un movimiento acelerado, así como el uso efectivo de funciones y bucles en Python.

Si bien estos resultados son alentadores, también eran esperables, ya que el desempeño de los estudiantes en la evaluación individual reflejó en gran medida el éxito obtenido en evaluaciones previas donde se aplicaron prácticas similares de PC al estudio del movimiento con velocidad constante. Sin embargo, esta última instancia no se limitó a replicar las tareas anteriores: presentó una nueva situación en la que los estudiantes debían interpretar de forma autónoma un código desconocido y vincularlo con un modelo físico. Esto exigió un mayor nivel de autonomía e integración conceptual. En este sentido, la evaluación individual constituyó una instancia significativa para valorar si los estudiantes eran capaces de transferir y aplicar sus conocimientos en un contexto novedoso, demostrando una competencia asociada a un aprendizaje más profundo.

En cuanto a posibles mejoras para futuras implementaciones de esta actividad, sería valioso incluir en la simulación una condición de detención que impida que el objeto continúe una vez que alcanza el suelo. Esto permitiría abrir una discusión sobre el papel y las limitaciones de los modelos matemáticos. Mientras que el código actual permite que la posición del objeto se vuelva negativa, la incorporación de esta condición podría enriquecer el valor pedagógico de la propuesta al invitar a los estudiantes a reflexionar sobre los supuestos detrás de las ecuaciones y cómo adaptarlas a escenarios más realis-

tas. En consonancia con este ajuste, la rúbrica también debería actualizarse para indicar que el nivel más alto de desempeño se alcanza cuando los estudiantes reconocen que el modelo deja de ser válido una vez que la posición del objeto se vuelve negativa.

Capítulo 6

Conclusiones y perspectivas

Este trabajo exploró la integración del PC en la enseñanza de la cinemática, a partir del diseño, implementación y evaluación de una secuencia didáctica desarrollada en un curso de física de educación media. A lo largo de la investigación buscamos responder tres preguntas principales: cómo puede diseñarse una secuencia que integre el PC con el estudio de la cinemática, qué características presentan los aprendizajes de los estudiantes en cinemática durante la implementación de la propuesta y qué prácticas de PC se manifestaron en sus producciones.

En relación con la primera pregunta, logramos diseñar una secuencia que articula de manera coherente los contenidos de cinemática con las prácticas de PC. El proceso de diseño se apoyó en la Ley General de la Cognición Extendida (da Rosa Zipitriá, 2018) y en el marco de prácticas de PC propuesto por Weller et al. (2022). Las actividades elaboradas buscaron favorecer el tránsito desde acciones concretas hacia representaciones computacionales más abstractas, promoviendo que los estudiantes construyeran conocimiento al pasar de la manipulación directa de tablas a la elaboración de programas y simulaciones. Este enfoque resultó adecuado para integrar la programación al estudio del movimiento, permitiendo que los algoritmos y las representaciones en Python se convirtieran en un medio para pensar la cinemática.

Respecto a la segunda pregunta, vinculada a los aprendizajes en cinemática, los resultados del TUGK mostraron mejoras significativas entre la aplicación inicial y final. Los estudiantes consolidaron la interpretación de las gráficas de posición y velocidad en función del tiempo, en particular la comprensión del concepto de pendiente y la distinción entre ambas magnitudes. Persistieron,

sin embargo, algunas dificultades en la relación entre velocidad y aceleración, lo que sugiere la necesidad de continuar trabajando en ese aspecto. En la evaluación mediante rúbricas observamos que los estudiantes lograron expresar con mayor claridad sus comprensiones cuando debían explicar o representar movimientos, en comparación con el formato de múltiple opción.

Más allá de estos avances conceptuales, el análisis de las producciones realizadas durante la secuencia permite identificar aprendizajes que difícilmente emergerían en un enfoque tradicional. Como se mostró en el capítulo anterior, las explicaciones de los estudiantes sobre el funcionamiento de los programas evidenciaron que pudieron establecer correspondencias explícitas entre expresiones matemáticas, procesos de cálculo y representaciones gráficas, promoviendo una comprensión más articulada de los modelos de movimiento. Asimismo, la elaboración de tablas de tiempo, posición y velocidad, realizadas de manera integrada con los programas y las simulaciones, difiere del modo en que habitualmente se construyen en enfoques tradicionales. Este trabajo favoreció que los estudiantes relacionaran directamente los valores numéricos con los comportamientos representados en las gráficas.

En conjunto, consideramos que los aprendizajes de cinemática alcanzados estuvieron dentro de lo esperable para este nivel educativo, pero la integración del PC ofreció oportunidades adicionales para construir y verificar modelos de movimiento, que difícilmente se presentan en enfoques exclusivamente analíticos o algebraicos. Si bien el estudio no contó con un grupo de control que permitiera comparar el impacto específico del trabajo con programación, los resultados obtenidos muestran que la inclusión del PC no obstruyó el aprendizaje de la cinemática y generó nuevas formas de exploración y comprensión de los fenómenos de movimiento.

En relación con la tercera pregunta, sobre las prácticas de PC involucradas, en las actividades desarrolladas estuvieron presentes diversas dimensiones de este tipo de pensamiento. Los estudiantes realizaron procesos de descomposición de problemas y de traducción de relaciones físicas a código, construyeron algoritmos y aplicaron lógica condicional. También hicieron uso de la generalización e incorporaron mayor complejidad a los modelos, eligieron distintas formas de representar y analizar datos, llevaron a cabo tareas de generación y manipulación de datos, así como de depuración, y trabajaron de forma colaborativa en torno a los modelos computacionales.

En cuanto a la evaluación, encontramos evidencias de desarrollo principal-

mente en algunas de estas prácticas: la traducción de relaciones físicas a código, la construcción de algoritmos, el uso de la lógica condicional y la generación de datos. Aunque el nivel de desempeño fue heterogéneo, observamos progresos claros en la comprensión del funcionamiento de los bucles iterativos y en la capacidad de interpretar el comportamiento de los programas en términos físicos.

El estudio presenta algunas limitaciones que deben considerarse. En primer lugar, la muestra fue reducida y perteneció a un único centro educativo, lo que impide generalizar los resultados. En segundo lugar, los estudiantes ya contaban con conocimientos básicos de programación, por lo que no podemos aislar con precisión el aporte específico de la secuencia de física en el desarrollo del PC. Además, la evaluación se basó principalmente en producciones escritas y programas, sin incluir observaciones sistemáticas de las interacciones grupales. Incorporar este tipo de información en futuras investigaciones permitiría enriquecer el análisis de las prácticas colaborativas y los procesos de depuración colectiva.

A partir de los resultados obtenidos, consideramos que existen varias líneas posibles de continuidad. Una primera consiste en profundizar el análisis de las relaciones entre física y programación, indagando cómo los estudiantes transfieren conocimiento entre ambas áreas y qué tipo de dificultades enfrentan en ese proceso. Otra línea de trabajo es extender la secuencia didáctica a otros contenidos de la física, particularmente aquellos que puedan abordarse mediante herramientas computacionales, y desarrollar instrumentos de evaluación más específicos para el razonamiento físico-computacional. También resulta necesario explorar formas de evaluación más integrales, que incluyan registros de aula y reflexiones de los propios estudiantes sobre su experiencia de programación. Asimismo, consideramos relevante elaborar materiales didácticos que faciliten la implementación de este tipo de actividades y que puedan adaptarse a distintos contextos educativos y niveles de experiencia en programación. Finalmente, entendemos que la sostenibilidad de propuestas de este tipo requiere avanzar en la formación docente orientada a la integración de PC y física, favoreciendo la apropiación de las herramientas computacionales y su uso pedagógico.

En conjunto, estas proyecciones permiten situar los aportes del presente trabajo en una perspectiva más amplia. En este sentido, la investigación desarrollada mostró que la incorporación del PC en la enseñanza de la física enriquece el aprendizaje conceptual cuando se realiza de manera articulada

con un curso de programación. Esta coordinación entre ambas áreas permitió que la escritura de código, la simulación y el análisis de datos se convirtieran en recursos para explorar fenómenos de movimiento y construir representaciones más integradas entre lo matemático, lo gráfico y lo computacional. Aunque los resultados son exploratorios, señalan el potencial de este enfoque para seguir profundizando la articulación entre la física y la programación en la educación media.

Referencias bibliográficas

- Agresti, A., y Finlay, B. (2009). *Statistical Methods for the Social Sciences* (4.^a ed.). Pearson Prentice Hall.
- Aho, A. V. (2012). Computation and computational thinking. *The computer journal*, 55(7), 832-835.
- Aiken, J. M., Caballero, M. D., Douglas, S. S., Burk, J. B., Scanlon, E. M., Thoms, B. D., y Schatz, M. F. (2013). Understanding student computational thinking with computational modeling. *AIP Conference Proceedings*, 1513(1), 46-49.
- Aiken, J. M., Caballero, M. D., Finkelstein, N. D., y Turpen, C. (2012). Modeling Instruction: Student Discourse and Reasoning. *Physical Review Special Topics - Physics Education Research*, 8(1), 010102. <https://doi.org/10.1103/PhysRevSTPER.8.010102>
- ANEPE. (2022, agosto). Marco Curricular Nacional 2022 [Versión 13]. <https://www.anep.edu.uy/sites/default/files/images/Archivos/publicaciones/Marco-Curricular-Nacional-2022/MCN%202%20Agosto%202022%20v13.pdf>
- Basu, S., Biswas, G., Sengupta, P., Dickes, A., Kinnebrew, J. S., y Clark, D. (2016). Identifying middle school students' challenges in computational thinking-based science learning. *Research and practice in technology enhanced learning*, 11, 1-35.
- Beichner, R. J. (1994). Testing student interpretation of kinematics graphs. *American journal of Physics*, 62(8), 750-762.
- Bocconi, S., Chiocciello, A., Kampylis, P., Dagienė, V., Wastiau, P., Engelhardt, K., Earp, J., Horvath, M., Jasuté, E., Malagoli, C., et al. (2022). Reviewing computational thinking in compulsory education: State of play and practices from computing education.
- Bollen, L., De Cock, M., Zuza, K., Guisasola, J., y van Kampen, P. (2016). Generalizing a categorization of students' interpretations of linear ki-

- nematics graphs. *Phys. Rev. Phys. Educ. Res.*, 12, 010108. <https://doi.org/10.1103/PhysRevPhysEducRes.12.010108>
- Brennan, K., y Resnick, M. (2012). New frameworks for studying and assessing the development of computational thinking. *Proceedings of the 2012 annual meeting of the American educational research association, Vancouver, Canada*, 1, 25.
- Brewe, E. (2008). Modeling Theory Applied: Modeling Instruction in Introductory Physics. *American Journal of Physics*, 76(12), 1155-1160. <https://doi.org/10.1119/1.2983148>
- Buffler, A., Lubben, F., Ibrahim, B., y Pillay, S. (2008). A Model-Based View of Science: Implications for Learning and Teaching. *South African Journal of Science*, 104, 351-355.
- Caballero, M. D., Burk, J. B., Aiken, J. M., Thoms, B. D., Douglas, S. S., Scanlon, E. M., y Schatz, M. F. (2014). Integrating numerical computation into the modeling instruction curriculum. *The Physics Teacher*, 52(1), 38-42.
- Caballero, M. D., Irving, P. W., y Merner, L. A. (2019). PICUP: Partnership for Integration of Computation in Undergraduate Physics. *Proceedings of the Physics Education Research Conference*, 80-83.
- Caballero, M. D., Kohlmyer, M. A., y Schatz, M. F. (2012). Implementing and Assessing Computational Modeling in Introductory Mechanics. *Physical Review Special Topics - Physics Education Research*, 8(2), 020106. <https://doi.org/10.1103/PhysRevSTPER.8.020106>
- Caballero, M. D., y Merner, L. (2018). Prevalence and nature of computational instruction in undergraduate physics programs across the United States. *Physical Review Physics Education Research*, 14(2), 020129.
- Caballero, M. D., y Odden, T. O. B. (2024). Computing in physics education. *nature physics*, 20(3), 339-341.
- Cabezas, M. (2024). Pensamiento Computacional, Educacion STEM y la Educacion Informatica: Cuestiones Pendientes. *Revista Reflexión e Investigación Educacional*, 45-58.
- Ceuppens, S., Bollen, L., Deprez, J., Dehaene, W., y De Cock, M. (2019). 9th grade students' understanding and strategies when solving $x(t)$ problems in 1D kinematics and $y(x)$ problems in mathematics. *Phys. Rev. Phys. Educ. Res.*, 15, 010101. <https://doi.org/10.1103/PhysRevPhysEducRes.15.010101>

- Chabay, R. W., y Sherwood, B. A. (2008). *Matter and Interactions* (2.ª ed.). Wiley.
- Chonacky, N., y Winch, D. (2008). Integrating computation into the undergraduate curriculum: A vision and guidelines for future developments. *American Journal of Physics*, 76(4), 327-333.
- Cohen, J. (2013). *Statistical power analysis for the behavioral sciences*. routledge.
- Crouch, C. H., y Mazur, E. (2001). Peer instruction: Ten years of experience and results. *American journal of physics*, 69(9), 970-977.
- da Rosa, S. (2015). The construction of knowledge of basic algorithms and data structures by novice learners. *PPIG*, 7.
- da Rosa, S., y Aguirre, A. (2018). Students teach a computer how to play a game. *Informatics in Schools. Fundamentals of Computer Science and Software Engineering: 11th International Conference on Informatics in Schools: Situation, Evolution, and Perspectives, ISSEP 2018, St. Petersburg, Russia, October 10-12, 2018, Proceedings* 11, 55-67.
- da Rosa, S., y Gómez, F. (2020). A research model in didactics of programming. *CLEI Electronic Journal*, 23(1), 5-1.
- da Rosa Zipitría, S. (2018). Piaget and computational thinking. *Proceedings of the 7th Computer Science Education Research Conference*, 44-50.
- Denning, P. J. (2017). Computational thinking in science. *American Scientist*, 105(1), 13-17.
- Denning, P. J., y Tedre, M. (2019). *Computational thinking*. Mit Press.
- Ding, L., Chabay, R., Sherwood, B., y Beichner, R. (2006). Evaluating an electricity and magnetism assessment tool: Brief electricity and magnetism assessment. *Phys. Rev. ST Phys. Educ. Res.*, 2, 010105. <https://doi.org/10.1103/PhysRevSTPER.2.010105>
- Docktor, J. L., y Mestre, J. P. (2014). Synthesis of Discipline-Based Education Research in Physics. *Physical Review Special Topics - Physics Education Research*, 10(2), 020119. <https://doi.org/10.1103/PhysRevSTPER.10.020119>
- Doran, R. L. (1980). *Basic Measurement and Evaluation of Science Instruction* [ERIC Document ED196733]. National Science Teachers Association.
- Duijzer, C., Van den Heuvel-Panhuizen, M., Veldhuis, M., Doorman, M., y Leseman, P. (2019). Embodied learning environments for graphing mo-

- tion: A systematic literature review. *Educational Psychology Review*, 31(3), 597-629.
- Dutra, M., Suárez, Á., y Martí, A. C. (2025). Code in Motion: Integrating Computational Thinking with Kinematics Exploration. *Aceptado en The Physics Teacher. arXiv preprint arXiv:2503.03850*.
- Fraser, J. M., Timan, A. L., Miller, K., Dowd, J. E., Tucker, L., y Mazur, E. (2014). Teaching and physics education research: Bridging the gap. *Reports on Progress in Physics*, 77(3), 032401.
- Gambrell, J. (2024). *Multiple Choice Computational Thinking Assessment for Introductory Physics* [Tesis doctoral, Drexel University].
- Gómez, F., y da Rosa, S. (2022). The construction of knowledge about programs. *PPIG*, 179-188.
- Hake, R. R. (1998). Interactive-engagement versus traditional methods: A six-thousand-student survey of mechanics test data for introductory physics courses. *American journal of Physics*, 66(1), 64-74.
- Hestenes, D. (1987). Toward a Modeling Theory of Physics Instruction. *American Journal of Physics*, 55(5), 440-454. <https://doi.org/10.1119/1.15129>
- Hutchins, N., Sayre, E. C., y Irving, P. W. (2020). Computational Modeling Activities for Learning Newton's Laws. *Physical Review Physics Education Research*, 16(2), 020136. <https://doi.org/10.1103/PhysRevPhysEducRes.16.020136>
- Hutchins, N., Biswas, G., Conlin, L., Emara, M., Grover, S., y Basu, S. (2018). Studying synergistic learning of physics and computational thinking in a learning by modeling environment. *Proceedings of the 26th International Conference on Computers in Education. Philippines: Asia-Pacific Society for Computers in Education*.
- Irving, P. W., Martinuk, M. S., y Sayre, E. C. (2017). Transitions in Students' Understanding of Energy in Quantum Physics. *Physical Review Physics Education Research*, 13(1), 010114. <https://doi.org/10.1103/PhysRevPhysEducRes.13.010114>
- Lave, J., y Wenger, E. (1991). *Situated Learning: Legitimate Peripheral Participation*. Cambridge University Press.
- Lodi, M., et al. (2020). Informatical thinking. *Olympiads in Informatics*, 14, 113-132.
- Lodi, M., y Martini, S. (2021). Computational thinking, between Papert and Wing. *Science & education*, 30(4), 883-908.

- McDermott, L. C., y Redish, E. F. (1999). Resource Letter: PER-1: Physics Education Research. *American Journal of Physics*, 67(9), 755-767. <https://doi.org/10.1119/1.19122>
- McDermott, L. C., Rosenquist, M. L., y Van Zee, E. H. (1987). Student difficulties in connecting graphs and physics: Examples from kinematics. *American journal of physics*, 55(6), 503-513.
- Meltzer, D. E., y Otero, V. K. (2015). *A Brief History of Physics Education Research*. American Association of Physics Teachers.
- Odden, T. O. B., y Caballero, M. D. (2023). Physics Computational Literacy: What, Why, and How? En M. F. Taşar y P. R. L. Heron (Eds.), *The International Handbook of Physics Education Research: Learning Physics* (pp. 19-1-19-28). AIP Publishing. https://doi.org/10.1063/9780735425477_019
- Odden, T. O. B., Lockwood, E., y Caballero, M. D. (2019). Physics computational literacy: An exploratory case study using computational essays. *Physical Review Physics Education Research*, 15(2), 020152.
- Orban, C., y Teeling-Smith, R. M. (2020). Computational Thinking in Introductory Physics. *Physical Review Physics Education Research*, 16(2), 020131. <https://doi.org/10.1103/PhysRevPhysEducRes.16.020131>
- Panthalookaran, V. (2018). Gamification of physics themes to nurture engineering professional and life skills. *2018 IEEE Global Engineering Education Conference (EDUCON)*, 931-939.
- Papert, S. (1980). Mindstorms: children, computers, and powerful ideas.
- Piaget, J. (1964). *La prise de conscience*. Presses Universitaires de France.
- Piaget, J. (1968). Genetic Epistemology (a series of lectures delivered by Piaget at Columbia University, translated by Eleanor Duckworth [Accessed: 2025-03-11]. <https://www.marxists.org/reference/subject/philosophy/works/fr/piaget.htm>
- Piaget, J. (1975). L'équilibration des structures cognitives. Problème central du développement.
- Piaget, J., y Garcia, R. (1989). *Psychogenesis and the history of science*. Columbia University Press.
- Plan Ceibal. (2022, abril). Fundamentos para el Desarrollo del Pensamiento Computacional en Educación Media. <https://pensamientocomputacional.ceibal.edu.uy/wp-content/uploads/2023/04/6298936761c84758304032.pdf>

- Plan Ceibal. (2023, junio). Marco Referencial de Pensamiento Computacional [Primera edición]. <https://pensamientocomputacional.ceibal.edu.uy/wp-content/uploads/2023/06/Compartido-Marco-Referencial-de-PC-1.pdf>
- Rose, J., O'Meara, J., Gerhardt, T., y Williams, M. (2016). Gamification: using elements of video games to improve engagement in an undergraduate physics class. *Physics Education*, 51(5), 055007.
- Skuse, B. (2019). The third pillar. *Physics World*, 32(3), 40.
- Sokoloff, D. R., Laws, P. W., y Thornton, R. K. (2007). RealTime Physics: active learning labs transforming the introductory laboratory. *European journal of physics*, 28(3), S83.
- Svensson, K., Eriksson, U., Lunk, B., Gustafsson, P., y Pendrill, A.-M. (2020). Student Difficulties and Potential Gains from Learning Computational Modeling in Physics. *European Journal of Physics*, 41(4), 045705. <https://doi.org/10.1088/1361-6404/ab8dbd>
- Trowbridge, D. E., y McDermott, L. C. (1980). Investigation of student understanding of the concept of velocity in one dimension. *American journal of Physics*, 48(12), 1020-1028.
- Trowbridge, D. E., y McDermott, L. C. (1981). Investigation of student understanding of the concept of acceleration. *American journal of Physics*, 49(3), 242-253.
- Weintrop, D., Beheshti, E., Horn, M. S., Orton, K., Jona, K., Trouille, L., y Wilensky, U. (2016). Defining Computational Thinking for Mathematics and Science Classrooms. *Journal of Science Education and Technology*, 25(1), 127-147. <https://doi.org/10.1007/s10956-015-9581-5>
- Weller, D. P., Bott, T. E., Caballero, M. D., y Irving, P. W. (2022). Development and illustration of a framework for computational thinking practices in introductory physics. *Physical Review Physics Education Research*, 18(2), 020106.
- Wing, J. M. (2006). Computational thinking. *Communications of the ACM*, 49(3), 33-35.
- Wing, J. M. (2008). Computational thinking and thinking about computing. *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences*, 366(1881), 3717-3725.
- Wing, J. (2010). Computational thinking: What and why?. The link-The Magazine of the Carnegie Mellon University School of Computer Science.

- Winsberg, E. (2019). *Science in the age of computer simulation*. University of Chicago Press.
- Zavala, G., Barniol, P., Tejeda y S. (2019). Evaluación del entendimiento de gráficas de cinemática utilizando un test de opción múltiple en español. *Revista Mexicana de Física E*, 65(2 Jul-Dec), 162-181. <https://doi.org/10.31349/RevMexFisE.65.162>
- Zavala, G., Tejeda, S., Barniol, P., y Beichner, R. J. (2017). Modifying the test of understanding graphs in kinematics. *Phys. Rev. Phys. Educ. Res.*, 13, 020111. <https://doi.org/10.1103/PhysRevPhysEducRes.13.020111>

ANEXOS

Anexo 1

Publicaciones y presentaciones en congresos derivadas de la tesis

A partir de este trabajo de tesis surgieron las siguientes publicaciones aceptadas en revistas arbitradas:

- Dutra, M., Suárez, Á., & Martí, A. (2025). *Code in motion: Integrating computational thinking with kinematics exploration*. The Physics Teacher.
- Dutra, M. 2025. *La computación como vía para superar obstáculos epistémicos en la enseñanza de la física*. Revista IRICE.

También se realizaron las siguientes presentaciones en congresos nacionales e internacionales (presentados por *):

- Mateo Dutra*, Álvaro Suárez y Arturo Martí. *Enhancing Kinematics Learning Through Computational Thinking and Real-Time Graphing*. Exposición oral presentada en la GIREP Summer School on Physics Education Research, GIREP, Países Bajos, 2025.
- Mateo Dutra* y Arturo Martí. *Dificultades conceptuales en cinemática y estrategias para enfrentarlas*. Póster presentado en el XVIII Encuentro de la Sociedad Uruguaya de Física, Uruguay, 2024.
- Mateo Dutra* y Arturo Martí. *Teaching sequences linking kinematics, functions and programming in Python*. Póster presentado en la XV Conferencia Interamericana en Educación en Física (CIAEF), Brasilia, 2023.

El videojuego presentado en la sección 3.3 fue el punto de partida del proyecto de investigación financiado por la ANII y Fundación Ceibal: *Dr. Simón dice: un entorno para aprender física integrando pensamiento computacional, sensores y juegos* (FSED_2_2023_1_179226). En este proyecto se desarrolló una versión web del videojuego con más funcionalidades, disponible en missionmotion.uy. En el marco de este proyecto, se realizaron las siguientes publicaciones aceptadas en revistas arbitradas:

- Dutra, M., Abreu, M., Monteiro, M., Sguilla, S., Stari, C., Suárez, A. & Martí, A. (2025). *Mejorando la comprensión de la cinemática a través de un videojuego basado en gráficos de movimiento en tiempo real*. Revista de Enseñanza de la Física.

También se realizaron las siguientes presentaciones en congresos nacionales e internacionales (presentados por *):

- Mateo Dutra*, Marcos Abreu, Martín Monteiro, Silvia Sguilla, C. Stari, Álvaro Suárez y Arturo Martí. *Enhancing Kinematics Understanding Through a Real-Time Graph-Based Motion Video Game*. Trabajo publicado y presentado en la conferencia internacional GIREP–EPEC on Physics Education Research, Leiden, 2025.
- Mateo Dutra*, Martín Monteiro*, Cecilia Stari* y Arturo Martí*. *Conferencia-taller: Mission Motion: enseñando cinemática con pensamiento computacional y sensores*. Presentada en el XXXV Encuentro Nacional de Profesores de Física, Uruguay, 2025.
- Mateo Dutra, Marcos Abreu, Martín Monteiro*, Silvia Sguilla, Cecilia Stari, Álvaro Suárez y Arturo Martí. *Mejorando la comprensión de la cinemática a través de un videojuego basado en gráficos de movimiento en tiempo real*. Trabajo presentado en la XVI Conferencia Interamericana sobre Educación en Física (CIAEF), Valparaíso, Chile, 2025.
- Mateo Dutra, Marcos Abreu, Martín Monteiro, Silvia Sguilla, Cecilia Stari, Álvaro Suárez y Arturo Martí*. *MissionMotion: un videojuego gratuito para enseñar cinemática en secundaria con sensores y gráficos en tiempo real*. Trabajo presentado en la Tercera Reunión Conjunta AFA–SUF, La Plata, Argentina, 2025.