



UNIVERSIDAD MICHOACANA DE SAN NICOLÁS DE HIDALGO

Facultad de Ciencias Físico Matemáticas

“Mat. Luis Manuel Rivera Gutiérrez”

**DISEÑO Y DESARROLLO DE UN AMBIENTE TECNOLÓGICO PARA
PROMOVER EL APRENDIZAJE DEL ÁLGEBRA, EN PARTICULAR LA
FACTORIZACIÓN**

TESIS

Autor

María Erika Cedeño Badillo

Licenciado en Ciencias Físico Matemáticas

Para obtener el grado de:

Asesor

Christian Morales Ontiveros

Maestro en Ciencias

Morelia Michoacán, abril de 2017

AGRADECIMIENTOS

A mi asesor, a quien le debo gran parte de mis conocimientos, gracias por su paciencia y enseñanzas.

M. C. Christian Morales Ontiveros

A mi madre por enseñarme a ser fuerte en los momentos difíciles.

Isabel Badillo Martínez

A mi esposo, por acompañarme en este proceso y sobre todo por su amor, comprensión, paciencia y fortaleza.

Ernesto Cuahutémoc Rodríguez García

A mi hijo, por su paciencia, alegría y porque tuvo que soportar largas horas sin la compañía de su mamá.

Rafael Ernesto Rodríguez Cedeño

A mis suegros, por su apoyo y su tiempo

Gloria García Cárdenas

Juan Ernesto Rodríguez Martínez

ÍNDICE

CAPÍTULO I: PLANTEAMIENTO DEL PROBLEMA	6
I.1. INTRODUCCIÓN	6
I.2. JUSTIFICACIÓN	8
I.3. OBJETIVOS	10
I.4. PREGUNTAS DE INVESTIGACIÓN	11
I.5. METODOLOGÍA	13
 CAPÍTULO II: MARCO TEÓRICO	 14
II.1. FUNDAMENTOS METODOLÓGICOS PARA EL DISEÑO DE SOFTWARE EDUCATIVO	 14
II.2. INCLUSIÓN DE AMBIENTES TECNOLÓGICOS PARA EL APRENDIZAJE DE LAS MATEMÁTICAS EN EL AULA	 22
II.3. USO DE ISETL COMO UNA HERRAMIENTA PARA APRENDER ÁLGEBRA ABSTRACTA	 27
II.4. ESTRATEGIAS PARA LA ENSEÑANZA DE LA ARITMÉTICA, EL ÁLGEBRA Y ALGUNOS CONCEPTOS RELACIONADOS A LOS SÍMBOLOS	 29
 CAPÍTULO III: PRODUCTOS NOTABLES Y FACTORIZACIÓN	 32
III.1. TÉRMINOS ALGEBRAICOS	32
III.2. PRODUCTOS NOTABLES	34
III.3. FACTORIZACIÓN DE POLINOMIOS	36

CAPÍTULO IV: DISEÑO Y DESARROLLO DEL SOFTWARE	38
IV.1. PRIMERA VERSIÓN DEL SOFTWARE	38
IV.2. MODELOS EN PAPEL	41
IV.3. IMPLEMENTACIÓN DE LA INTERFACE Y FUNCIONAMIENTO DE LA APLICACIÓN	43
IV.4. ANÁLISIS DEL CÓDIGO	79
 CAPÍTULO V: CONCLUSIONES	 90
 BIBLIOGRAFÍA	 92

RESUMEN

Existen varios trabajos de investigación que muestran ciertas dificultades en los estudiantes que cursan la materia de álgebra en el nivel medio superior y superior, particularmente hablando del tema de la factorización. Esas dificultades implican varias cuestiones de fondo en la asimilación de conceptos. La factorización es uno de los temas fundamentales del álgebra en el que se espera que el estudiante domine ciertos conceptos y sea capaz de manipularlos para, de la generalización llegar a la abstracción y viceversa. En este trabajo se presenta el desarrollo de un software educativo, el cual incluye varios escenarios, diferentes representaciones de equivalencia, diferentes construcciones, sobre posición de objetos, de tal manera que el estudiante pueda aprender incluso de los errores. Este software permite preparar al estudiante en el camino a la factorización.

Palabras clave: *Álgebra, Factorización, Representación, Tecnología, Ambientes de aprendizaje.*

ABSTRACT

There are several research papers that show certain difficulties in the students who study the subject of algebra in the upper and upper middle level, particularly speaking of the topic of factorization. These difficulties imply several fundamental questions in the assimilation of concepts. Factorization is one of the fundamental themes of algebra in which the student is expected to master certain concepts and be able to manipulate them, from generalization to abstraction and vice versa. This work presents the development of an educational software, which includes several scenarios, different representations of equivalence, different constructions, on position of objects, so that the student can learn even from mistakes. This software allows to prepare the student on the way to factorization.

CAPÍTULO I

PLANTEAMIENTO DEL PROBLEMA

I.1. INTRODUCCIÓN

La tecnología ha venido evolucionando en los últimos años y ésta es cada vez más utilizada en la vida cotidiana a tal grado que incluso en la Reforma Educativa (SEP, 2015) se manifiesta dentro de los beneficios concretos, que se puede esperar:

- “Los alumnos reforzarán su educación integral con mayor tiempo en las aulas para hacer tareas deportivas, artísticas, de aprendizaje de idiomas y manejo de las tecnologías de la información y la comunicación”.
- “Los planteles educativos contarán con mejores instalaciones y tecnología moderna”.

Existen varios trabajos de investigación que muestran ciertas dificultades en los estudiantes que cursan la materia de álgebra en el nivel medio superior y algunos de los primeros niveles de educación superior, particularmente hablando del tema de la factorización. Esas dificultades implican varias cuestiones de fondo en la asimilación de conceptos, como pueden ser la manipulación aritmética, la interpretación del lenguaje simbólico, así como la memorización de reglas o fórmulas para resolver problemas algebraicos que se les presentan y de las cuales muchas veces, terminan por no saber con exactitud cuál es la regla o fórmula que se debe aplicar.

En la actualidad existen programas muy sofisticados, robustos y eficientes para la enseñanza de las matemáticas, pero uno de los problemas que se presentan en esta área es que no siempre se encuentra software específico que atender los problemas conceptuales, sumado el hecho de que el sistema educativo no ha logrado incorporar todo ese impulso tecnológico que se ha desarrollado, es por ello que en las aulas se siguen utilizando métodos

antiguos por la falta de la incorporación de la tecnología, ya que al no ser aprovechada hace que la metodología para el proceso de enseñanza aprendizaje sea la tradicional.

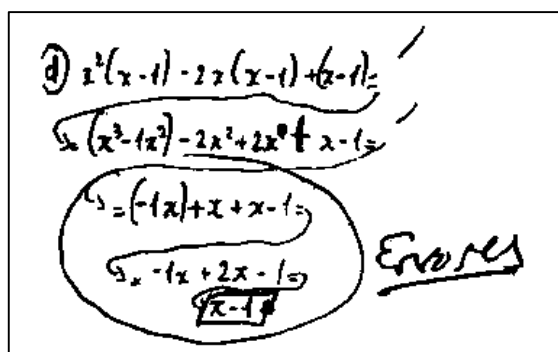
Por otra parte, los profesores deben buscar alternativas de enseñanza de forma que se logre atraer aún más la atención del estudiante, es por ello que al desarrollar el software se pretende que se acompañe de actividades preparadas por el profesor que estén dentro del programa académico al que se pretenda incorporar, para que el profesor pueda adaptar e incorporar en su clase este tipo de herramienta tecnológica en temas que son para algunos alumnos difíciles de comprender.

En ese sentido, se necesitan herramientas tecnológicas que se adapten a las necesidades de los estudiantes y que éstas sean tanto atractivas como efectivas en su uso.

I. 2. JUSTIFICACIÓN

En la etapa escolar el álgebra es considerada como una de las partes de la matemática que influye en el aspecto formativo, por la potencia y simplicidad de sus registros formales y por sus métodos, pero su aprendizaje genera muchos problemas a los alumnos, éstos tienen que ver con la complejidad de los objetos del álgebra, con los procesos de pensamiento algebraico, con el desarrollo cognitivo de los alumnos, con los métodos de enseñanza e incluso con actitudes afectivas y emocionales hacia el álgebra. Estos problemas de distinta procedencia, se conectan y refuerzan en redes complejas que se reducen en la práctica en forma de obstáculos y se manifiestan en los alumnos, mediante errores (Medina, 1998).

Algunos de los errores más comunes que cometen los estudiantes son al sumar, restar, multiplicar, factorizar y al transponer términos en las expresiones algebraicas. En Saucedo (2007) presentan esta falta cometida (ver Figura 1), en la cual se muestra que el alumno no sabe operar con polinomios, ya que no encuentra una justificación para lo realizado por él mismo.



The image shows a student's handwritten work on a math problem. At the top, the expression $x^2(x-1) - 2x(x-1) + (x-1)$ is written. Below it, the student has written $x(x^2-1x^2) - 2x^2 + 2x^0 + x-1$, which is circled. Underneath that, the student has written $= (-1x) + x + x - 1 =$, also circled. Below this, the student has written $3x - 1x + 2x - 1 =$, with $3x - 1x$ circled. To the right of the work, the word "Errores" is written in a stylized, slanted font.

Figura 1. El alumno no sabe operar con polinomios

En Thomas, Ramos, & Maldonado (2014), analizan algunos de los errores de mayor incidencia cometidos por a los alumnos de álgebra I de la Universidad Pedagógica Nacional Francisco Morazán al aplicarles una prueba evaluativa (ver Figura 2 y 3), los cuales fueron los siguientes:

The image shows a handwritten equation: $\frac{x^3+8}{-x^2+4} = \frac{(x+2)(x^2+2x+4)}{(x-2)(x+2)} = \frac{(x^2+2x+4)}{(x-2)} = \frac{(x+2)^2}{(x-2)}$. The denominator in the second step, $(x-2)(x+2)$, is circled in red, indicating a sign error where $-x^2+4$ was incorrectly factored as $(x-2)(x+2)$ instead of $-(x-2)(x+2)$.

Figura 2. Error en el denominador.

Se observa el error en la manipulación del signo que lleva la expresión del denominador.

The image shows a handwritten equation: $\frac{1-b^2}{(1+bx)^2-(b+x)^2} = \frac{(1+b)(1-b)}{((1+bx)+(b+x)) - ((1+bx)-(b+x))}$. The denominator is circled in red, indicating an error in the expansion of $(1+bx)^2$ and $(b+x)^2$. The student incorrectly wrote $(1+bx)^2$ as $(1+bx) + (b+x)$ and $(b+x)^2$ as $(1+bx) - (b+x)$.

Figura 3. Error en el desarrollo del binomio.

El estudiante cometió la falta al tratar de resolver el producto notable en el denominador.

Es por ello que existe la necesidad de crear un software que ayude en el aprendizaje y la manipulación de elementos algebraicos, en la suma, resta, multiplicación y factorización de polinomios de segundo grado, de tal manera que su aprendizaje sea dinámico.

I.3. OBJETIVOS

El presente trabajo de tesis consiste en el diseño y creación de un software educativo para la enseñanza y el aprendizaje de la factorización algebraica, que generalmente se encuentra situada en la asignatura de álgebra tanto en el nivel medio superior como en el nivel básico, dicho software será programado totalmente en el lenguaje de programación conocido como Java. Por lo que, se desprenden los siguientes:

OBJETIVO GENERAL

- Diseño y desarrollo de un software dinámico para la enseñanza de la factorización utilizando el concepto de área de una figura regular.

OBJETIVOS PARTICULARES

- Diseño, desarrollo e implementación de una interface amigable en el lenguaje de programación Java, para la manipulación de objetos matemáticos.
- Interacción dinámica entre los objetos matemáticos a través del uso del ratón de la computadora.
- Realizar prueba técnica de la estabilidad del software.

I. 4. PREGUNTAS DE INVESTIGACIÓN

- 1. ¿Qué tipo de objetos matemáticos serían los más apropiados a utilizar al interior de un software dinámico, que coadyuven en el aprendizaje de la descomposición de expresiones algebraicas?**

Consideramos ésta como una pregunta fundamental en el desarrollo del software educativo a desarrollar, puesto que de acuerdo al marco teórico con el que se desarrolla esta propuesta es necesario darle un sentido más simple a la aritmética y a las expresiones algebraicas, para lo cual se requiere de objetos que se puedan manipular.

- 2. ¿Cuál sería el potencial del software para promover el desarrollo de ideas matemáticas en el aprendizaje de la factorización de expresiones algebraicas?**

De acuerdo a la metodología detrás del software a desarrollar, existen varias herramientas manipulativas en físico que han aportado significativamente en el desarrollo y habilidades matemáticas en los estudiantes que los han utilizado, sin embargo, se parte desde la perspectiva optimista de que si esos trabajos físicos han logrado lo antes citado, con mayor razón, si se tratara de un software que contuviera todas esas herramientas físicas, así como actividades que lleven al estudiante a apropiarse de las bases fundamentales de la aritmética.

- 3. ¿Cuál debería ser la mejor estrategia de interacción entre los objetos matemáticos del software y la organización de estos, para beneficiar el aprendizaje significativo de la factorización?**

En la actualidad al momento de usar tecnología, prácticamente en la mayoría de los casos, estamos acostumbrados al arrastre de objetos ya sea con el puntero del mouse o en el caso de las tabletas o Smartphone, el arrastre de objetos usando los dedos de la mano, bajo esta idea creemos importante y fundamental que nuestro software debe incluir ese tipo de manipulación en los objetos, pero ésta debe ir guiada de tal suerte que no se confunda entre

solo manipular objetos por juego o a prueba y error, a asimilar con más conciencia las bases elementales de los objetos matemáticos que se muestran en el software.

I.5. METODOLOGÍA

El desarrollo de software educativo implica muchas cuestiones, entre ellas el tener un equipo de desarrolladores programadores, pedagogos, así como de diseñadores de imágenes, con el fin de que el software sea aceptado y robusto.

En ese sentido, se requiere tener una metodología básica para desarrollar software educativo efectivo, por lo cual en el presente trabajo se utiliza como base la metodología sugerida por Clements & Battista (2000) para el diseño de software educativo. Si bien, en el presente trabajo no se llevaron a cabo todas las fases de la metodología, (las cuales se describen en el siguiente capítulo), si se llevaron a cabo las más fundamentales.

CAPÍTULO II

MARCO TEÓRICO

II.1. FUNDAMENTOS METODOLÓGICOS PARA EL DISEÑO Y DESARROLLO DE SOFTWARE EDUCATIVO

No hay duda que la tecnología tendrá un gran impacto en la enseñanza y el aprendizaje de las matemáticas y la ciencia. Sin embargo, sostenemos que los resultados de las empresas encargadas del diseño de software educativo no han estado a la altura de su potencial, al no incorporar elementos didácticos que les permitan producir software educativo efectivo, en el sentido de ser simultáneamente pedagógico y computacionalmente eficaz.

En esta sección se describe un modelo para la investigación integrada del software y del desarrollo de planes de estudio propuesto por Clements y Battista (2000), que ayudará a cumplir con el potencial del desarrollo de software efectivo.

El diseño de software debe tener un fundamento teórico explícito y empírico, más allá de su origen en alguna comprensión intuitiva del aprendizaje de los estudiantes, debe interactuar con el continuo desarrollo de la teoría y la investigación de largo alcance hacia la idea de experimentar una teoría probando el software. Para maximizar sus contribuciones en la matemática educativa, la teoría, la investigación y el desarrollo de software deben tener un gran número de temas en consideración y proceder a través de una variedad de fases.

Fase 1: Diseño de los objetivos iniciales

Objetivos de aprendizaje matemático. La primera fase comienza con la identificación de una problemática dominante de las matemáticas. Este dominio debería ser significativo de dos maneras: el aprendizaje del dominio debe contribuir de manera significativa al desarrollo matemático de los estudiantes y contribuir sustantivamente a la investigación y la teoría.

En el establecimiento de metas de aprendizaje matemático, se debe analizar las recomendaciones de la reforma educativa, así como aspectos socioculturales y técnicos, ya que se espera que los usuarios finales sean estudiantes que puedan participar en investigaciones de campo.

El resultado de esta primera fase es la descripción detallada de un aspecto problemático de las matemáticas, sobre todo si el objetivo es el desarrollo innovador de software y si las matemáticas difieren del uso común.

Fase 2: Construcción un modelo explícito de conocimiento y aprendizaje de los estudiantes, basado en los objetivos de la fase 1

El siguiente paso es construir un modelo de aprendizaje de los estudiantes, que sea suficientemente explícito para describir los procesos involucrados en la construcción de los objetivos matemáticos. Con frecuencia, los modelos existentes están disponibles, aunque varían en grado de especificidad; algunos modelos altamente especializados se pueden utilizar directamente, mientras que otros son generales y, por lo tanto, la mayoría sugerentes. Especialmente en este último caso, el desarrollador utiliza entrevistas clínicas y observaciones para examinar los conocimientos de los estudiantes del dominio de contenido, incluyendo los conceptos, estrategias e ideas intuitivas utilizadas para resolver problemas. Los estudiantes pueden usar papel y materiales físicos (aunque estos pueden ser configurados para anticipar las experiencias de software que se diseñarán).

En estos experimentos, el profesor trata de crear una situación o tarea que provocará conceptos y procesos pertinentes. El entrevistador hace preguntas de seguimiento para conocer con el mayor detalle posible qué y cómo los estudiantes están pensando en la situación. Las parejas de niños pueden observarse trabajar en problemas, ya que pueden ser más relajados y expresivos de su pensamiento.

Una vez que el modelo se ha desarrollado parcialmente, se prueba y se amplía con la enseñanza exploratoria. Este enfoque contrasta con el de la perspectiva "maravillosa de

ideas", con su enfoque inicial en lo que el equipo puede hacer y tener éxito en los intentos de utilizar estas capacidades tecnológicas para enseñar matemáticas.

Si se han previsto actividades específicas como un componente del software, técnicas similares se utilizan para evaluar la idoneidad relativa de los problemas propuestos. Es decir, debe haber enseñanza que es exploratoria e intuitiva, seguido de un análisis. Los modelos mentales de los investigadores sobre el pensamiento de los estudiantes tienen el objetivo de guiar su diseño de actividades. Como sucede en cada fase del desarrollo, más experiencias de los investigadores con los nuevos estudiantes proporcionan retroalimentación que conduce a la modificación de ese modelo mental.

El resultado final de esta fase, es un modelo explícito del aprendizaje de las matemáticas de los estudiantes. Las actividades y resultados de esta fase están conectados, incluso superpuestos, con los de las siguientes fases, ya que el modelo de aprendizaje de los estudiantes se debe desarrollar constantemente.

Fase 3: Creación de un diseño inicial de software y de las actividades

Planificar el diseño. Nuestras descripciones e ilustraciones de esta fase se vuelven más específicas. Continuamos nuestro enfoque en el diseño de micromundos basados en la investigación teórica. Sin embargo, hay muchas otras posibilidades, tales como la producción de:

- Una herramienta cognitiva, es decir, una representación de las operaciones de nivel inferior obviando la necesidad de los novatos para lograr la automatización de estas habilidades antes de la realización de pensamiento de orden superior.
- Una simulación de fenómenos.
- Un lenguaje de programación especializado.
- Escritura de componentes modulares que puedan representar cualquiera de estos tipos.

Con base en el modelo de aprendizaje de los estudiantes generado en la Fase 2, los desarrolladores crean un bosquejo básico para describir los objetos que van a constituir el entorno de software y las acciones que se pueden realizar con ellos. Estas acciones en los objetos deben reflejar la actividad matemática supuesta de los estudiantes en la mayor medida posible. El diseño también describe cómo las acciones deben ser instanciadas. La justificación de estas acciones y todas las otras decisiones de diseño, es fundamental para el desarrollo y el proceso de investigación.

El diseño basado en objetos y acciones, obligan al desarrollador a centrarse en acciones o procesos explícitos y lo que van a decir a los estudiantes. Esta característica refleja el beneficio atribuido a modelos de la ciencia cognitiva del pensamiento humano; que no permiten "cajas negras" para ocultar las debilidades de la teoría. Hay que destacar que los diseños no están determinados plenamente por esta línea de razonamiento: la intuición y el arte de enseñar juegan un papel crítico en el diseño de software.

Considerar la posibilidad de características únicas de la computadora. Los desarrolladores deben tener en cuenta las posibles características únicas de la computadora en la planificación de los objetos y acciones. Características adicionales concebibles de manuales informáticos incluyen:

- Flexibilidad.
- Cambio de arreglo o representación.
- Almacenamiento y recuperación de configuraciones más adelante.
- Grabar y reproducir las acciones de los estudiantes.
- Vinculación de lo concreto y lo simbólico con retroalimentación.
- Vincular dinámicamente múltiples representaciones.
- Centrar la atención y aumentar la motivación.
- Cambio de la naturaleza del manipulador.

Además de las características antes mencionadas, las siguientes se han identificado como contribuciones únicas:

- Promover la conexión de las representaciones formales con representaciones visuales dinámicas, el apoyo a la construcción de estrategias y de ideas matemáticas, intuiciones iniciales y enfoques visuales.
- Fomentar la manipulación de formas específicas, de manera que ayuden a los estudiantes a verlos como representantes matemáticos de una clase de formas.
- Alentar a preguntarse acerca de las cosas y plantear problemas, al proporcionar un entorno en el que se ponen a prueba las ideas y se recibe retroalimentación sobre ellas.
- Facilitar la reflexión de los estudiantes sobre la modificación de su representación mental.
- Proporcionar flexibilidad suficiente para la gama de oportunidades para explorar las matemáticas, se amplía para los estudiantes y los profesores.

Realizar un esquema de las actividades. La creación del bosquejo de tareas de instrucción esperadas y de las actividades preliminares para el uso del software deben basarse en:

- El modelo de aprendizaje de los estudiantes.
- Una revisión de la literatura profesional.
- Las propias experiencias del pasado de uno mismo.

Fase 4: Evaluar los componentes del software

Durante esta fase, los componentes del software se ponen a prueba mediante entrevistas clínicas y observaciones de un pequeño número de estudiantes. Las preguntas críticas que deben responderse en esta fase son: ¿Los usuarios son capaces de controlar el software con el dispositivo de entrada? ¿Cómo los usuarios interactúan y comprenden el diseño de la pantalla, los objetos y las acciones? El objetivo es entender el significado que los estudiantes dan a los objetos y acciones que se han creado para tratar de promover su aprendizaje.

En esta fase se puede usar papel o materiales físicos, como maquetas del software o versiones de prototipo tempranos. Algunos desarrolladores pueden utilizar el software que puede crear prototipos rápidamente, antes de programar en un lenguaje de bajo nivel más rápido. En esta y la siguiente fase, la comunicación entre el diseñador y el desarrollador (si son diferentes personas), es esencial. Si la programación se lleva a cabo por separado, la completa comunicación entre todos los aspectos (metas, acciones, objetos, estética, etc.) deben de garantizarse.

Fase 5: Evaluar prototipos y currículo

En esta fase se continúa con la evaluación del software en su versión prototipo. El objetivo es evaluar si las características del ambiente computacional diseñado, corresponden a los modelos de pensamiento de los estudiantes, a través de sus acciones sobre los objetos y su actividad matemática. A falta de esta correspondencia, se deberá reformular ya sea el modelo mental del estudiante o la forma en que dicho modelo es desarrollado a través del software.

También se evalúan aspectos pedagógicos, poniendo atención en las interpretaciones que dan los estudiantes a la retroalimentación que les proporciona el software. En este sentido, puede ser útil un experimento de enseñanza con un grupo pequeño de estudiantes. Generalmente una exploración libre del software, precede a la introducción de actividades. Esta información servirá para construir modelos más refinados, además de contar con datos empíricos que apoyen la realización de investigaciones con respecto a la interacción de los estudiantes con la computadora.

Fase 6: Realizar pruebas piloto en un aula

Hay dos ejes de investigación durante esta fase. En primer lugar, los experimentos de enseñanza continúan, pero en una forma diferente. Se lleva a cabo una experimentación de enseñanza en el aula con estudiantes, donde el objetivo es dar sentido a las actividades

curriculares que fueron plasmadas en el software y ver si éstas son entendidas por los estudiantes.

En segundo lugar y al mismo tiempo, se observa toda la clase de información sobre la utilidad, eficacia del software y el plan de estudios. Se presta atención en cómo las experiencias del software refuerzan, complementan y extienden las experiencias de aprendizaje con objetos manipulativos o impresos.

Fase 7: Pruebas de conducta de campo en múltiples aulas

Estas pruebas de campo se llevan a cabo con varios profesores que no están conectados inicial e íntimamente con el desarrollo del software. Una vez más, se observa toda la clase de información sobre la eficacia y facilidad de uso del software y el plan de estudios, pero se pone más énfasis en la usabilidad por los profesores. Queremos saber si el software y sus materiales de apoyo son lo suficientemente flexibles para soportar múltiples situaciones (por ejemplo, la variación en el número de computadoras disponibles), varios modos de instrucción (por ejemplo, la demostración de una clase, la clase de discusión, el trabajo en grupos pequeños), y diferentes modos y estilos de trabajo.

En esta fase el tiempo es a menudo más extenso, por ejemplo, hay que dar a los estudiantes períodos largos, para internalizar o construir procesos mentales que reflejan las herramientas de software. La evaluación de la competencia de los estudiantes y el potencial de aprendizaje, debe incluir la interacción extendida, el trabajo ampliado sobre las actividades y puede requerir compromisos repetidos con el software. Si los proyectos son un aspecto del plan de estudios, el tiempo adicional es especialmente importante. Además, las interacciones se vuelven complejas; los investigadores deben tener en cuenta la actividad, objetivos personales, dónde y cuándo los estudiantes se vuelven perdidos, cómo (o si) se pueden recuperar, y así sucesivamente.

En resumen, las fases 4 a 7 proporcionan un enfoque integral para obtener el asesoramiento de los usuarios y los datos de investigación importantes. No todos los

proyectos pueden o deben emplear cada fase: sin embargo, las razones de la omisión de cualquiera de las fases y la coherencia de las mismas que se incluirán, deberán ser consideradas cuidadosamente.

Fase 8: Repetición

Es esta fase se realizan acciones interactivas y recursivas, dentro y entre las fases anteriores. Sin embargo, a menudo se hace un progreso sustantivo cuando un proyecto completo es revisado, refinado, reconceptualizado y recreado.

Fase 9: Publicar el software producido

Esta fase parece obvia. El software y las actividades de estudio pueden ser difundidas a través de una variedad de canales, de las editoriales comerciales o internet. Es importante la publicación de la investigación, tanto sumativa como formativa. Se han creado muchas piezas interesantes de software, sin embargo, la experiencia desarrollada durante la producción del software no siempre se difunde.

II.2. INCLUSIÓN DE AMBIENTES TECNOLÓGICOS PARA EL APRENDIZAJE DE LAS MATEMÁTICAS EN EL AULA

Las herramientas informáticas, nos permiten abordar las matemáticas desde diferentes ángulos de lo que se hace tradicionalmente. Muchos aspectos sobre los procesos dinámicos se pueden describir de manera cualitativa más que cuantitativa. Las herramientas informáticas pueden cambiar la calidad de los objetos matemáticos y los procesos en la experiencia de aprendizaje del alumno, éstas pueden convertirse en herramientas cognitivas.

Un diseñador de herramientas informáticas puede tener el poder para decidir sobre el tipo y la presentación de los gráficos; la visualización simultánea de la información numérica y gráfica; la introducción de sofisticadas representaciones, vínculos entre las dos representaciones, acoplamiento y desacoplamiento de las representaciones, y así sucesivamente. Las herramientas informáticas introducen en la educación matemática nuevas oportunidades, así como también nuevos problemas que deben resolverse (Dreyfus, 1994).

OPORTUNIDADES

Una de las oportunidades más mencionadas ofrecidas por herramientas computacionales, son su potencial para el uso múltiple de representaciones ligadas. La idea es utilizar varias representaciones del mismo concepto, es que los diferentes aspectos de éste se resalten en diferentes representaciones y que a los estudiantes les ayude a conectar conceptualmente diferentes aspectos sobre las representaciones correspondientes, Duval (2006) nos dice:

“El software proporciona herramientas para mostrar “instantáneamente” tantas representaciones diferentes como sean necesarias. Por tanto, los estudiantes también pueden obtener el rango de posibles representaciones de los objetos con los que están trabajando o que están usando como herramientas. Además, el software puede dar una percepción dinámica de la transformación de representación frente al soporte estático del papel. Tareas específicas como

traducir o transferir también pueden servir para preparar a los estudiantes para que reaccionen ante una clase de representación dada (verbal, simbólica o visual) cambiándola por otra.”

El uso de múltiples representaciones para apoyar la adquisición de conceptos científicos y matemáticos, es una parte común entre los diseñadores de software, ya que los estudiantes son capaces de integrar la información en un todo coherente.

Una de las razones por lo que las computadoras han aumentado el potencial de múltiples representaciones es la infografía, que hacen posible las representaciones esquemáticas. Incluso sin estar necesariamente vinculados a otras representaciones, el razonamiento con representaciones esquemáticas ha recibido recientemente mucha atención de los investigadores.

Las computadoras hacen posible representar visualmente las matemáticas, por medio de diagramas, con una cantidad de estructura no ofrecido por cualquier otro medio. Las representaciones graficas en una pantalla de los objetos matemáticos y sus relaciones, permite una acción directa sobre estos objetos y la observación de los cambios en las relaciones de la forma esquemática; esto puede ayudar al estudiante a darse cuenta de la existencia y comprender la naturaleza de las relaciones (Schwarz & Dreyfus, 1993).

Una nueva oportunidad basada en herramienta para la enseñanza de las matemáticas, se debe a la posibilidad de dejar que los ordenadores hagan los "cálculos triviales" como la repetida aplicación de la función en el ejemplo de los procesos dinámicos. La idea es que los estudiantes puedan operar a un alto nivel conceptual; en otras palabras, que puedan concentrarse en las operaciones que se piensan para ser el centro de atención y dejar las operaciones de nivel inferior a la computadora. Por ejemplo, al aprender manipulación algebraica, se pueden dejar cálculos numéricos a la computadora.

DIFICULTADES

La misma posibilidad, que fue presentado en el párrafo anterior como una oportunidad, también puede ser visto como la causa de un problema. Dejando cálculos numéricos al ordenador durante las actividades que tienen como objetivo el aprendizaje sobre la manipulación algebraica, puede ser considerados como un paso en una secuencia de niveles ordenados jerárquicamente:

- Aprender acerca de los números.
- Automatizar cálculos numéricos, para su uso en el aprendizaje de álgebra.
- Automatizar manipulaciones algebraicas, para su uso en el aprendizaje del cálculo.
- Automatizar la integración, para su uso en el aprendizaje de ecuaciones diferenciales.
- Automatizar la solución de ecuaciones diferenciales, para su uso en el aprendizaje dinámico.

Centrar la atención en un problema que puede surgir cuando los estudiantes están utilizando herramientas computacionales con tales capacidades jerarquías, ya que no podemos evitar que los estudiantes también usen la computadora para hacer el álgebra mientras se supone que deben estar aprendiendo manipulaciones algebraicas. Más tarde en la vida, que se espera que tengan un sistema de álgebra computacional a su disposición siempre que lo necesitan.

Por otro lado, el desarrollo del currículo y los maestros deben continuar la enseñanza, por lo tanto, tienen que tomar decisiones teniendo dos opciones: una es intentar desarrollar material curricular adecuado para su uso con un sistema de álgebra computacional general e investigar los efectos; la otra opción, es diseñar herramientas informáticas específicas para su uso en entornos educativos.

Herramientas basadas en las matemáticas contra herramientas basadas en la didáctica

Una separación se hace evidente, las herramientas basadas en matemáticas como los sistemas de álgebra computacional y hojas de cálculo, están contruidos para ajustarse la lógica interna y la estructura del área de contenido, respetan el orden lógico y la estructura inseparable al área de contenido matemático.

Por otra parte, los estudiantes pueden llegar a ser muy aptos para el uso de derivadas o integrales en la herramienta basada en las matemáticas que han aprendido acerca de ellas, pero ni siquiera reconocen estos conceptos fuera de la herramienta, la transferencia conceptual es notoriamente débil.

Esta herramienta puede no ser didácticamente apropiada en el sentido de que es compatible con la ejecución de procedimientos, mientras deja de lado la estructura conceptual profunda. Una herramienta matemática basada en la capacidad de llevar a cabo cálculos y dibujar gráficos de manera muy eficiente, por lo general no se tendrá en cuenta ninguna de las dificultades conceptuales que se derivan para el estudiante que se enfrenta con la construcción de una imagen mental adecuada, por ejemplo, la noción de límite o derivada.

Estas herramientas tienen como objetivo la creación de experiencias de aprendizaje que promuevan la construcción progresiva por parte del alumno de imágenes del concepto flexibles, amplias y aplicables, de nociones tales como la relación, la función derivada, etc. Un objetivo de la construcción de tales imágenes del concepto, es la flexibilidad en la resolución de problemas.

Por otro lado, si la adquisición de conceptos locales es el principal objetivo de un plan de estudios, la herramienta basada en la didáctica puede ser la elección correcta. Pero precisamente esta misma característica, es un problema de la herramienta basada en la didáctica: pueden ser demasiado local, diseñado específicamente y adaptado a un concepto particular o grupo de conceptos, o de un plan de estudios en particular. Una herramienta

basada en la didáctica, puede ser diseñada para ser adaptado a un determinado plan de estudios con sus objetivos específicos de aprendizaje. Se convierte en un componente orgánico de ese plan de estudios.

Desde el punto de vista de un profesor en matemáticas, los conceptos matemáticos deben ser el objetivo primordial y debe determinar los instrumentos. Las matemáticas que se enseñan en la escuela, podrían llegar a ser más orientadas a la herramienta, al mismo tiempo, las herramientas de propósito general pueden ser didácticamente más adecuadas.

II.3. USO DE ISETL COMO UNA HERRAMIENTA PARA APRENDER ÁLGEBRA ABSTRACTA

El lenguaje de programación interactivo **ISETL** pertenece a un paradigma de lenguajes de programación, llamado lenguaje de programación matemática y se caracteriza por poseer una sintaxis similar a la notación matemática estándar. En la Universidad del estado de Tennessee (MTSU), realizaron cambios en el currículo de álgebra abstracta para incluir el uso de esta herramienta, esto implicó realizar actividades informáticas y aprendizaje cooperativo dentro del aula. Los resultados de incorporar esta herramienta se muestran a continuación (Krishanamani & Kimmons, 1994):

Los cambios: Todo el currículo de álgebra abstracta fue rediseñado, para incorporar métodos interactivos constructivos que implican actividades informáticas y aprendizaje cooperativo. Los cambios introducidos resultaron en los descubrimientos de los estudiantes. Las actividades informáticas permitieron que el profesor introdujera conceptos con bastante antelación a lo que de otro modo podrían haber sido introducidos y les permitía a los estudiantes estar más familiarizados con las ideas, abstractos y de otro tipo. Se desvió de "estudiar sólo para las pruebas" a un patrón de aprendizaje continuo y tener discusiones con compañeros y profesores. Se sintieron cómodos al hacer predicciones. La retención mejoró y los estudiantes estuvieron orgullosos de presentar sus descubrimientos.

Observaciones: Los estudiantes mostraron una mejor comprensión del concepto de un grupo, cuantificadores, negación de una proposición y función. ISETL los ayudó en el pensamiento lógico.

Preocupaciones del estudiante: Los estudiantes tenían muchas preocupaciones sobre este nuevo método, sobre el tiempo que se debe invertir en este curso, las pruebas, entre otras.

Pasos para reducir la ansiedad: Se permitió cierta flexibilidad en la formación de grupos de aprendizaje cooperativo. El paso más importante fue la línea directa de tareas. Se

dedicó mucho tiempo a ayudar a los estudiantes, especialmente en la computadora. Una semana especial fue asignada como semana abstracta de la conciencia del álgebra. Las actividades tales como las sesiones de computadora y las presentaciones de los estudiantes fueron los catalizadores de estas interacciones.

Rendimiento: Las actuaciones impresionantes de algunos estudiantes merecen una mención especial. Estos talentos habrían pasado desapercibidos en un curso tradicional.

Reconocimiento: El desempeño de los estudiantes fue destacado, ellos se encontraban entusiasmados al presentar sus propios resultados. Algunos estudiantes asumieron roles de liderazgo que normalmente no habrían hecho. La prueba de cada estudiante fue personalizada en apariencia. Cada estudiante consideró que sus respuestas eran importantes. La discusión de las metas y el desempeño se hizo sobre una base individual.

Se necesitan nuevas ideas y soluciones para hacer frente a los desafíos que se encuentran en el uso de la tecnología, los profesores universitarios necesitan probar la tecnología y hacer la investigación con diferentes métodos de enseñanza. Esto nos permitirá desarrollar formas efectivas de utilizar la tecnología para ayudar a los estudiantes a aprender álgebra.

II.4. ESTRATEGIAS PARA LA ENSEÑANZA DEL LA ARITMÉTICA, EL ÁLGEBRA Y ALGUNOS CONCEPTOS RELACIONADOS A LOS SÍMBOLOS

Una de los temas que preocupan en la didáctica de la matemática es el de la enseñanza y aprendizaje del álgebra, ya que el cambio del pensamiento aritmético al pensamiento algebraico tiene su grado de dificultad e implica que el estudiante cambie su modo de pensar.

Este cambio hace que los estudiantes cometan errores cuando quieren pasar del lenguaje natural que es el que hasta ahora han utilizado, al lenguaje algebraico; incorporado el problema de tipo didáctico por parte del docente, el cual se basa en la memorización excesiva de reglas, procedimientos, automatización del trabajo algebraico (Arreaza & Carvajal, 2013).

Para fomentar en los estudiantes un aprendizaje mediante el descubrimiento y la reflexión podemos utilizar las representaciones geométricas ya que ayudan a visualizar cualquier propiedad, representación simbólica y conceptos que nos ofrezca el álgebra para luego pasar a su representación formal o abstracta.

Uno de los materiales manipulativos que permiten al estudiante construir su propio aprendizaje son los Bloques de Dienes, este material consiste de varios cuadrados grandes y pequeños, de rectángulos de ciertas dimensiones elaborados en dos colores, uno representa las cantidades positivas y el otro las cantidades negativas, son un recurso que facilita la creación de un ambiente agradable y permite que el estudiante se sienta motivado hacia la actividad de aprender y conocer. Aun teniendo este tipo de materiales el docente necesita tener las siguientes consideraciones:

- Planificación de las actividades educativas, concretando objetivos didácticos y contenidos.
- El docente debe actuar como mediador en el proceso educativo.
- El estudiante debe construir los conocimientos o significados de los objetos matemáticos, partiendo de la naturalidad que les brindan los materiales concretos,

para luego pasar al formalismo matemático utilizando el lenguaje y la simbología matemática.

- Estimular el trabajo en grupo, donde los estudiantes puedan aprender de sus compañeros y corregir entre ellos sus errores.

Entre los contenidos que se pueden trabajar con los bloques de Dienes son: conjunto de los números naturales, conjunto de los números enteros, suma, resta, multiplicación, división y factorización de polinomios hasta de grado 2, así como algunos elementos de la teoría de números.

En Kieran (1981), se examina las investigaciones recientes acerca de los usos del signo igual entre estudiantes de preescolar, primaria y secundaria, así como en los estudiantes universitarios.

El símbolo que se utiliza para demostrar equivalencia no siempre es interpretado en esos términos por parte del alumno, ya que una interpretación de equivalencia del signo igual no parece venir con facilidad o rapidez para muchos estudiantes.

Los niños en edad preescolar tienen la noción de saber si dos conjuntos son numéricamente iguales, es porque con ellos se obtienen el mismo número cardinal cuando cuenta cada conjunto y para determinar si éstos son diferentes, puede considerar una idea comparativa de la igualdad al contar dos diferentes conjuntos y comparar sus cantidades.

En el período de la educación primaria los niños tienen conocimientos acertados sobre la suma y la resta. Muchos niños aprenden con bastante rapidez a leer y escriben el simbolismo elemental de la aritmética simple, pero no necesariamente lo entienden, ellos lo interpretan en términos de las acciones a realizar.

Para muchos alumnos de 13 años de edad, se encuentran en un período de transición entre pedir la respuesta después del signo igual y aceptar el signo igual como símbolo de equivalencia. Durante este período, existe una gran confusión donde los alumnos cometen

errores y una de las dificultades en el uso de problemas de palabras para dar significado a los símbolos matemáticos, es el tema de las relaciones frente a los procedimientos.

Las comparaciones que los sujetos eran capaz de hacer entre los lados izquierdo y derecho del signo igual, era la de sugerir que el símbolo de la igualdad estaba siendo visto más como símbolo relacional que como una "señal de hacer algo".

La razón para extender la noción del signo igual para incluir múltiples operaciones en ambos lados, era proporcionar una base para la construcción posterior de lo que significan las ecuaciones algebraicas no triviales. Si esta expansión no se realiza en primer lugar, el estudiante seguiría trayendo con él en el estudio de ecuaciones algebraicas la idea de que el resultado está siempre en el lado derecho del signo igual.

Siguiendo los modos de representación enactiva, icónico y simbólicas, el estudiante puede adquirir una comprensión intuitiva del significado de la ecuación y luego gradualmente transformar éste a una comprensión de la forma de una ecuación algebraica.

Al igual que con las identidades aritméticas, el lado derecho de una ecuación algebraica no tiene que contener la respuesta, sino que más bien podría ser alguna expresión que tenía el mismo valor que el lado izquierdo. Sin embargo, la capacidad de considerar una ecuación algebraica como una expresión de equivalencia porque ambas partes no parecen tener el mismo valor, sea suficiente para una adecuada conceptualización del proceso de resolución de ecuaciones. No sólo resolver la ecuación implicar una comprensión de la noción de que la derecha y la izquierda de los lados de la ecuación son expresiones equivalentes, sino también que cada ecuación puede ser sustituida por una ecuación equivalente.

Se puede argumentar que estos estudiantes poseen una noción de equivalencia profunda del signo igual y que no son más que accesos directos en su procedimiento de trabajo, los cuales generan errores y sugieren que muchos estudiantes de secundaria todavía interpreten el signo igual como un símbolo operador.

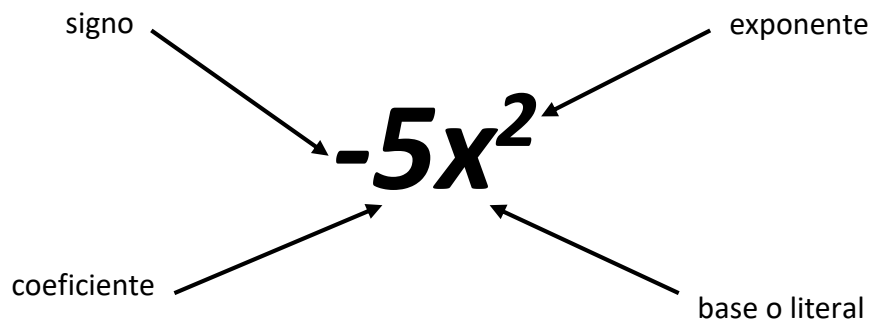
CAPÍTULO III

PRODUCTOS NOTABLES Y FACTORIZACIÓN

III.1. TÉRMINOS ALGEBRAICOS

Se denomina *término* a toda *expresión algebraica* cuyas partes no están separadas por los signos + o -.

En todo término algebraico pueden distinguirse cuatro elementos: el signo, el coeficiente, la parte literal y el grado.



Signo

Los términos que van precedidos del signo + se llaman términos positivos, en tanto los términos que van precedidos del signo - se llaman términos negativos. El signo + se acostumbra omitir delante de los términos positivos; así pues, cuando un término no va precedido de ningún signo se sobreentiende de que es positivo.

Coeficiente

Se llama coeficiente al número o letra que se le coloca delante de una cantidad para multiplicarla. El coeficiente indica el número de veces que dicha cantidad debe tomarse como sumando. En el caso de que una cantidad no vaya precedida de un coeficiente numérico se sobreentiende que el coeficiente es la unidad.

Parte literal

La parte literal está formada por las letras que haya en el término.

Grado

El grado de un término con respecto a una letra es el exponente de dicha letra.

La expresión algebraica que consta de un solo término es llamada monomio, la suma o resta de dos monomios origina un binomio, la de tres un trinomio, y en general las de dos o más términos con exponentes enteros positivos se denominan *polinomios*.

III.2. PRODUCTOS NOTABLES

Los siguientes productos de polinomios o productos notables, son muy utilizados en álgebra, éstos cumplen ciertas reglas fijas y ayudan en el proceso de factorización de polinomios. Cada producto notable corresponde a una fórmula de factorización. Por ejemplo, la factorización de una diferencia de cuadrados perfectos es un producto de dos binomios conjugados.

Cuando en una multiplicación de polinomios, los términos están en desordenados, bastará con reordenarlos para identificar si se trata de algún tipo de producto notable, para posteriormente realizar las operaciones correspondientes. En las siguientes definiciones, sean A, B y C monomios cualesquiera tal que sumándolos o restándolos se obtienen los binomios A+B, A-B, A+C, etc.

Factor común:

$$\pm C (\pm A \pm B) = \pm CA \pm CB$$

El producto de un monomio por un binomio es igual a la suma (resta) del monomio por el primer término del binomio, más (menos) el monomio por el segundo término del binomio.

Ejemplo:

$$3 (x + 2) = 3x + 6$$

Binomio al cuadrado:

$$(A \pm B) (A \pm B) = (A \pm B)^2 = A^2 \pm 2AB + B^2$$

El cuadrado de la suma (o resta) de dos monomios es igual al cuadrado del primer monomio más (menos) el doble producto del primero por el segundo más el cuadrado del segundo monomio.

Ejemplo:

$$(-1 + 2x)^2 = (2x - 1)^2 = 4x^2 - 4x + 1$$

Producto de dos binomios conjugados:

$$(A + B)(A - B) = A^2 - B^2$$

El producto de la suma de dos monomios por la diferencia de los mismos es igual al cuadrado del primer monomio menos el cuadrado del segundo monomio.

Ejemplo:

$$(-3 + x)(x + 3) = (x - 3)(x + 3) = x^2 - 9$$

Producto de dos binomios con un término común:

$$(\pm A \pm B)(\pm A \pm C) = A^2 \pm (\pm B \pm C)A \pm BC$$

El producto de dos binomios con término común es igual al cuadrado del primer término, o término común, más (menos) el producto de la suma (resta) de los dos segundos términos por el término común, más (menos) el producto de los segundos términos.

Ejemplos:

- $(2 + y)(y + 4) = (y + 2)(y + 4) = y^2 + (2 + 4)y + 8 = y^2 + 6y + 8$
- $(-3 - x)(-x + 2) = (-x - 3)(-x + 2) = x^2 - (-3 + 2)x - 6 = x^2 + x - 6$

III.3. FACTORIZACIÓN DE POLINOMIOS

Si un polinomio es producto de otros polinomios entonces a cada uno de los polinomios anteriores se les denomina factores del polinomio original. El proceso de encontrar los factores de un polinomio se conoce como factorización o descomposición del polinomio. La descomposición de un polinomio consiste en expresarlo como producto de otros polinomios de igual o menor grado que él mismo. Existen diversas técnicas que se pueden utilizar según sea la forma de la expresión que se vaya a factorizar.

Factorización por factor común:

Se aplica en binomios, trinomios y polinomios de cuatro términos o más. El factor común es aquello que se encuentra multiplicando en cada uno de los términos del polinomio. Se debe de observar los coeficientes de los términos extrayendo el máximo común denominador de ellos, de la base o literal se extrae la de menor exponente. Se escribe el factor común seguido de un paréntesis donde se anota el polinomio que queda después de que el factor común ha sido factorizado.

$$\text{Ejemplo: } 3x^2 + 6xy = 3x (x + 2y)$$

Factorización por diferencia de cuadrados:

Se aplica solamente en binomios donde el primer término es positivo y el segundo término es negativo, los coeficientes de los términos tienen raíz cuadrada exacta y la base de los términos tienen exponentes pares. Se debe extraer la raíz cuadrada de cada término, se abren dos grupos de paréntesis que se multiplican, las raíces cuadradas se anotan dentro de cada paréntesis, en uno se anotan sumándose y en el otro restándose, obteniéndose como resultado un binomio conjugado.

$$\text{Ejemplo: } 9x^2 - 4y^2 = (3x+2y) (3x-2y)$$

Factorización de trinomio cuadrado perfecto:

El trinomio debe de estar organizado en forma ascendente o descendente; el primer y el tercer término deben ser positivos y deben tener tanto el coeficiente como la parte literal de ambos términos raíz cuadrada exacta. Debe de cumplir también que el segundo término del trinomio sea el doble producto de las raíces del primer y tercer término, sin importar el signo.

Si se verifica que se trata de un trinomio cuadrado perfecto entonces construye un binomio al cuadrado anotándose las raíces cuadradas del primer y tercer término, entre ellas el signo del segundo término.

$$\text{Ejemplo: } x^2 - 4xy + 4y^2 = (x - 2y)^2$$

Factorización de trinomio cuadrado de la forma $x^2 + bx + c$:

El trinomio debe estar organizado de manera descendente y el coeficiente del primer término debe de ser 1, y el grado del exponente del primer término debe ser el doble del grado del segundo término.

Se abren dos grupos de paréntesis y se anota al comienzo de cada uno la raíz cuadrada del primer término, posteriormente se definen los signos, el signo del primer paréntesis se obtiene al multiplicar los signos del primer y segundo término; el signo del segundo paréntesis se obtiene al multiplicar los signos del segundo y tercer término. Se buscan dos cantidades que multiplicadas den como resultado el término independiente c, y que sumadas den como resultado el coeficiente del segundo término b., y se anotan en los espacios restantes de los paréntesis.

$$\text{Ejemplo: } x^2 - 2x - 15 = (x - 5)(x + 3)$$

CAPÍTULO IV

DISEÑO Y DESARROLLO DEL SOFTWARE EDUCATIVO

IV.1. PRIMERA VERSIÓN DEL SOFTWARE

En los cursos de Diseño y Desarrollo de Software Educativo 1 y 2, impartidas en la Facultad de Ciencias Físico Matemáticas, se originaron varias ideas que dieron como resultado un pequeño prototipo de aplicación. En ella se exponía una sección para representar polinomios en los cuales se mostraba un ejercicio fijo y bloques predeterminados (ver Figura 4), el usuario tenía que acomodarlos en el cuadrado rojo y ajústalo para formar un rectángulo.

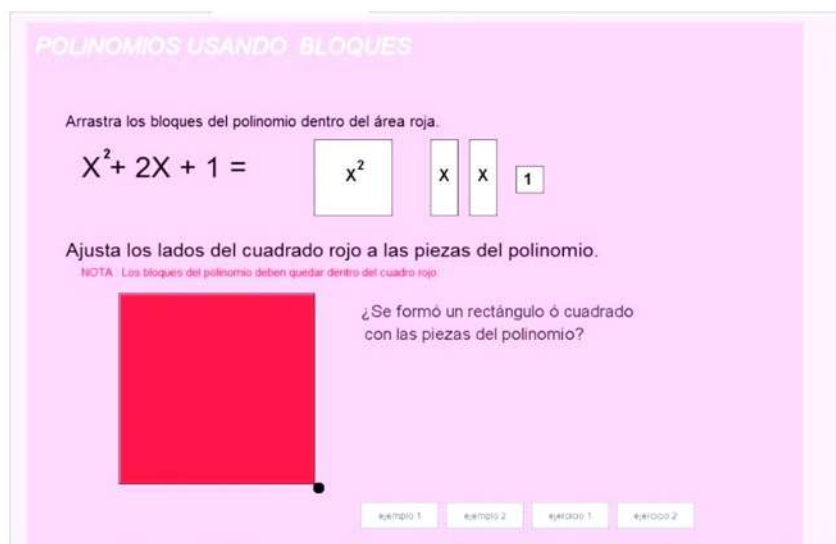


Figura 4. Primera sección con un ejercicio fijo.

En estos ejercicios se pretendía que el usuario observara al acomodar los bloques que para algunos ejercicios se tendrían que colocar otros bloques complementarios y así formar un rectángulo (ver Figura 5).

POLINOMIOS USANDO BLOQUES

Arrastra los bloques del polinomio dentro del área roja.

$X^2 + (-4) =$

x^2 -1 -1 -1 -1

Ajusta los lados del cuadrado rojo a las piezas del polinomio.

NOTA: Los bloques del polinomio deben quedar dentro del cuadro rojo.

¿Se formó un rectángulo ó cuadrado con las piezas del polinomio?

¿Qué bloques tendrías que agregar para completar el cuadrado sin alterar el polinomio?

Figura 5. Ejercicio para formar un cuadrado con los bloques.

En otra sección del programa se mostraba un ejercicio en el que además de los bloques que representaban al polinomio, había otros bloques para completar el rectángulo (ver Figura 6).

FACTORIZACION DE POLINOMIOS

Mueve los bloques del polinomio al cuadro rojo.

$X^2 + (-4) =$

x^2 -1 -1 -1 -1

Crea los bloques que hagan falta para completar el cuadrado.

NOTA: Los bloques del polinomio deben quedar dentro del cuadro rojo.

Usa algunas de estas áreas para completar el rectángulo

$2X$ -2 2 $-2X$ 1

¿Cuales son las medidas del rectángulo formado?

Base: Altura:

Figura 6. Ejercicio con algunas opciones de bloques para la respuesta.

Posteriormente se pedía que se escribiera la base y altura del rectángulo formado con los bloques (ver Figura 7).

FACTORIZACION DE POLINOMIOS

Mueve los bloques del polinomio al cuadro rojo.

$$x^2 + (-4) =$$

Crea los bloques que hagan falta para completar el cuadrado.

NOTA Los bloques del polinomio deben quedar dentro del cuadro rojo.

Usa algunas de estas áreas para completar el rectángulo

¿Cuales son las medidas del rectángulo formado?

Base: $x+2$ Altura: $x-2$

$$(x + 2)(x - 2) = x^2 - 4$$

Figura 7. Ejercicio para factorizar un polinomio.

En esta versión no se disponía de una estructura inicial sobre los conceptos en el tema de factorización gráfica, ni variedad en los ejercicios, por ello se siguió trabajando.

IV.2. MODELOS EN PAPEL

Al analizar la primera aplicación reconociendo sus fallas, así como su potencial, surgieron las siguientes ideas (ver Figura 8):

1. El software tenía que dividirse en varias secciones, las cuales partieran de lo sencillo a lo complejo.
2. Se tendría que colocar un menú para elegir el ejercicio.

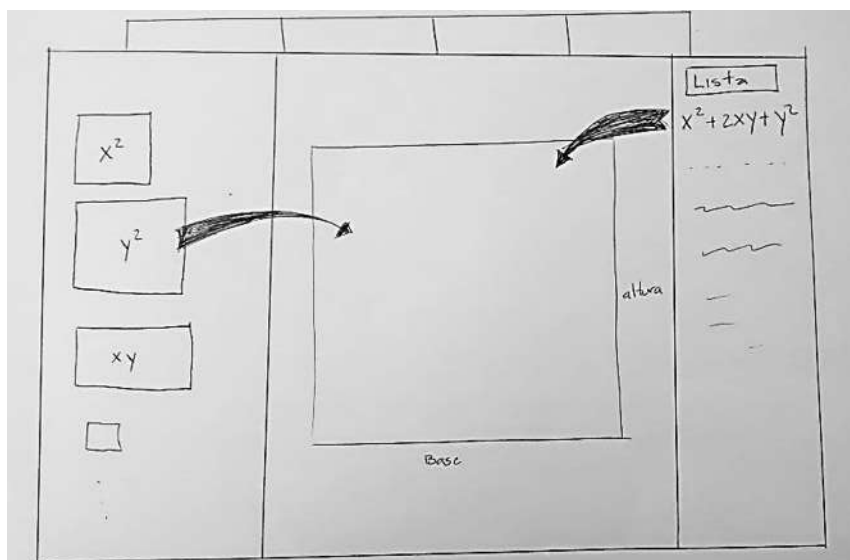


Figura 8. Primer prototipo en papel.

3. Se deberían colocar más de un bloque de cada tipo y todos se deberían poder arrastrar al área central, llevando en alguna variable el conteo de cada uno.
4. Se debería poder visualizar al mismo tiempo una representación gráfica y una representación algebraica de los ejercicios.

5. Los bloques tendrían que poder cambiar el tamaño de sus lados, para resaltar el concepto de variable tanto de X como de Y.
6. Para optimizar espacio dentro de la interfaz los bloques positivos y negativos tendrían que ser un mismo bloque, es decir, al activar algo tendría que cambiar de positivo a negativo y viceversa (ver Figura 9).

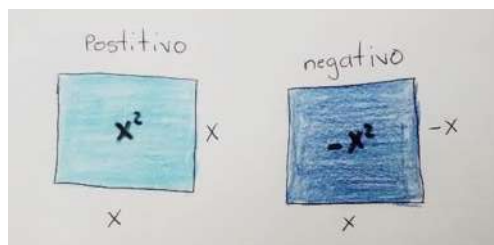


Figura 9. Componentes en papel.

7. Se pretendía incorporar el cambio de orientación de los bloques para poder acomodarlos dentro de cierta área y poder armar un rectángulo.
8. Para las secciones de Factorización se quería dar una ayuda visual o guías, para que el alumno pudiera acomodar fácilmente los bloques en cierta área y después factorizar el ejercicio sin ayuda en secciones posteriores.

IV.3. IMPLEMENTACIÓN DE LA INTERFACE Y FUNCIONAMIENTO DE LA APLICACIÓN

Al iniciar la aplicación se muestra la sección de representación de números enteros, las siguientes pestañas superiores contienen las partes de suma, resta, multiplicación, factorización de casos generales y factorización libre de polinomios, como se muestra en la Figura 10.



Figura 10. Presentación Inicial de la aplicación.

A la izquierda de la ventana se encuentran ubicadas **las herramientas** necesarias para la realización de los ejercicios, en la parte central se encuentran dos áreas, el área **Representación gráfica** y el área de **Representación algebraica**. En el área de **Representación gráfica** se colocarán los bloques necesarios para completar el ejercicio, cambia su color cuando el ejercicio se completó correctamente. En la parte de **Representación algebraica** se mostrará la expresión algebraica de los bloques que son arrastrados al área gráfica mediante etiquetas.

En la ventana se encuentra un recuadro con la leyenda “**Selecciona un ejercicio**” el cual, al presionar la flecha, desplegará un panel con botones para seleccionar el ejercicio.

A continuación, se describen las partes que componen el programa, así como su funcionamiento, posteriormente se describirán las secciones y subsecciones del mismo.

LAS HERRAMIENTAS

En la parte izquierda de la aplicación se encuentran las herramientas o bloques utilizados en todas las secciones (ver Figura 11), en las partes de representación, suma y resta de enteros sólo se encuentran los bloques de área 1 y los bloques de área entera variable, que llamaremos ab , ya que sólo se usarán valores enteros en estas secciones.

En las demás secciones que involucran operaciones con polinomios se encuentran todos los bloques X^2 , Y^2 , X , Y , XY , 1 , ab .

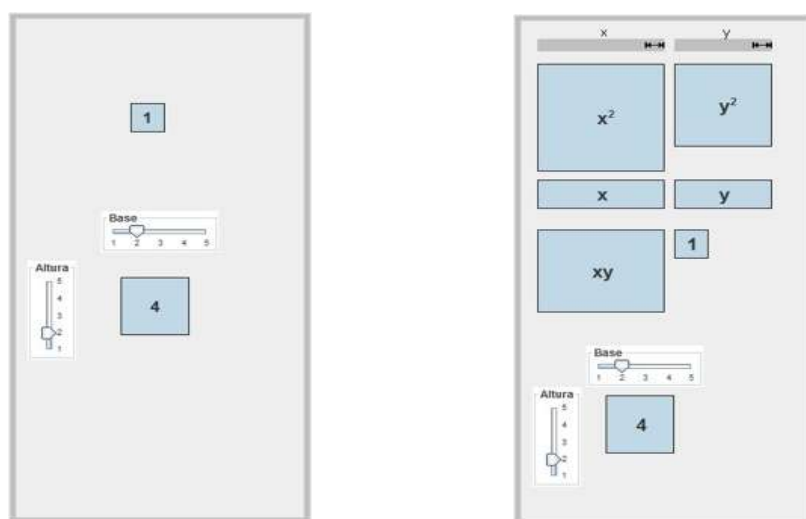


Figura 11. Herramientas de las secciones del software.

Dimensiones de los bloques:

El nombre de los bloques representa el área del mismo, es decir, la multiplicación de la base por la altura. El bloque de área X^2 tiene como base y altura X , esta magnitud es variable. El bloque de área Y^2 tiene como base y altura Y , esta magnitud es variable. El bloque de área X tiene como base X y altura 1, la altura es fija y la base es variable. El bloque de área Y tiene como base Y y altura 1, la altura es fija y la base es variable. El bloque de área XY tiene como base X y altura Y , la altura y la base son variables. El bloque de área 1 tiene como base y altura 1, la altura y la base son constantes (ver Figura 12).

Cambio de tamaño:

En la parte superior de las herramientas se encuentran dos deslizadores, con título X y Y respectivamente, los cuales sirven para modificar el tamaño de los lados X y Y de los bloques (ver Figura 12). El bloque de área 1 se mantiene constante en todo el programa.

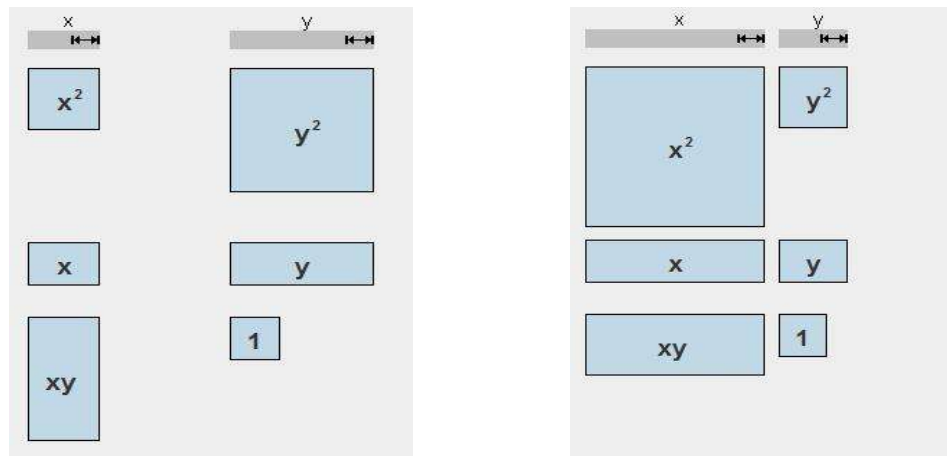


Figura 12. Cambio de las dimensiones de los bloques.

Cambio de signo:

Todos los bloques, excepto el bloque ab , pueden cambiar de positivo a negativo y viceversa al dar doble clic dentro del bloque. Modificando su color y la etiqueta de su nombre (ver Figura 13).

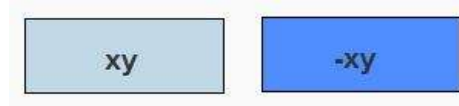


Figura 13. Cambio de signo del bloque xy .

Cambio de orientación:

Los bloques que pueden cambiar su orientación son: Y , X y XY (ver Figura 14). Para rotarlos se tendrá que mantener presionada la tecla *shift* del teclado y dar un clic dentro del bloque.

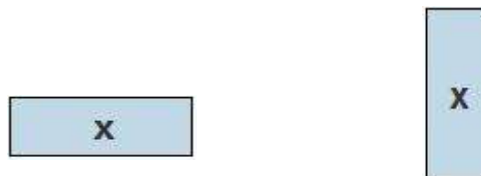


Figura 14. Cambio de orientación del bloque x .

Cambio de área del bloque ab :

En la parte inferior de las herramientas se encuentra el bloque ab de área variable, al iniciar la aplicación se presenta con área de 4 unidades, base 2 y altura 2, éstas dimensiones se pueden modificar usando los deslizadores altura y base localizados a la derecha y arriba del bloque ab respectivamente.

La base y la altura del bloque ab puede variar de 1 a 5, es decir, si se elige como base 3 y altura 3, el área del bloque ab sería de 9 unidades (ver Figura 15).



Figura 15. Cambio de dimensión del bloque ab.

BOTONES DE LAS SECCIONES

Cada sección tiene los siguientes botones:

Reinicio: Al presionarse regresa todas las variables, imágenes y valores a su configuración inicial (ver Figura 16).



Figura 16. Botón de reinicio general.

Uso de herramientas: El usuario puede ver una serie de imágenes con indicaciones para el funcionamiento de los bloques o herramientas, como el cambio de tamaño, cambio de signo y cambio de orientación (ver Figura 17).

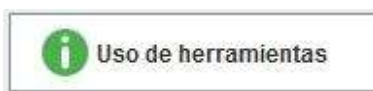


Figura 17. Botón con información sobre las herramientas.

Ejemplo: Muestra una galería de imágenes sobre cómo resolver un ejercicio de la sección donde se encuentre (ver Figura 18).



Figura 18. Botón para visualizar un ejemplo.

Botón ok: se localiza en los paneles para seleccionar un ejercicio, al presionarlo, el usuario confirma la selección del término del polinomio (ver Figura 19).



Figura 19. Botón ok.

Botón flecha: se localiza en la parte de “selecciona ejercicio” o “Altura x Base” (ver Figura 20), al presionarse despliega los paneles para la elección de los elementos del ejercicio que se requiera en la sección donde se encuentre.



Figura 20. Botón flecha.

PANEL PARA LA SELECCIÓN DE EJERCICIOS

Selección de ejercicio con números enteros:

En las secciones que involucran solamente el uso de números enteros, se presenta el panel para elegir un numero entero, la etiqueta “a” representa el número entero que se deberá modificar.

Se presenta inicialmente el valor 1 en el campo de entrada, este valor se puede cambiar usando los botones pequeños de flecha, seleccionando un valor entero para “a” de una secuencia ordenada con rango de -10 a 10 (ver Figura 21). Al elegir el entero el usuario deberá presionar el botón ok.

Figura 21. Sección para elegir enteros.

Selección de ejercicio con polinomios:

Estos paneles muestran en la parte superior etiquetas las cuales tomarán el valor que el usuario determine, a la izquierda un campo de entrada para la elección del coeficiente entero, semejante al de elección de números enteros, pero en algunas secciones de la aplicación el rango es diferente. Al centro del panel se encuentran seis botones para la elección de la parte literal (ver Figura 22). En la izquierda se encuentra un *botón ok* que finaliza la elección del término del polinomio, coloreándolo de negro y mostrando un nuevo panel para la elección del siguiente término según sea el caso.

Figura 22. Sección para elegir un trinomio.

Cada término del polinomio a elegir se muestra con un color diferente al igual que los botones del panel, en un polinomio no se pueden elegir dos o más términos con la misma parte literal ya que al realizar una elección el botón de esa variable se inhabilita.

Ejemplo en la selección de un trinomio:

1. Elegimos el coeficiente y la parte literal del primer término que se presentan en color verde (ver Figura 23), después presionamos el botón ok.

3 y +b xy+c y ²							▼
3	x	y	xy	x ²	y ²	a	✓

Figura 23. Elección del primer término del trinomio.

2. Elegimos el coeficiente y la parte literal del segundo término que se presentan en color morado (ver Figura 24), después presionamos el botón ok.

3 y -2 x +c y ²							▼
-2	x	y	xy	x ²	y ²	a	✓

Figura 24. Elección del segundo término del trinomio.

3. Elegimos el coeficiente y la parte literal del tercer término que se presentan en color naranja (ver Figura 25), después presionamos el botón ok.

3 y -2 x +10							▼
10	x	y	xy	x ²	y ²	a	✓

Figura 25. Elección del tercer término del trinomio.

4. Al terminar, los paneles desaparecen y sólo se muestra la etiqueta con el trinomio seleccionado en color negro (ver Figura 26).

3 y -2 x +10							▼
--------------	--	--	--	--	--	--	---

Figura 26. Elección total del trinomio.

Selección de base y altura:

En la sección de multiplicación, factorización de casos generales y factorización libre se encuentran los paneles para seleccionar la base y altura de alguna figura rectangular que represente algún polinomio, como se muestra en la Figura 27, los cuales funcionan de la misma manera que los paneles para seleccionar ejercicio. Estos pequeños paneles solamente tienen tres botones en la parte central.



Figura 27. Sección para elegir base y altura de una figura rectangular.

SECCIONES DE LA APLICACIÓN

REPRESENTACIÓN DE POLINOMIOS

La pestaña de *representación* tiene cuatro secciones ubicadas en las pestañas laterales: números enteros, monomios, binomios y trinomios (ver Figura 28).

Representación de números enteros:

Al iniciar el usuario deberá ubicarse en el panel “selecciona un ejercicio” y elegir un número entero, después deberá arrastrar los bloques necesarios al área de Representación gráfica, si la representación es correcta el recuadro donde se colocaron los bloques cambiará de color (ver Figura 29).



Figura 28. Sección para representar números enteros.

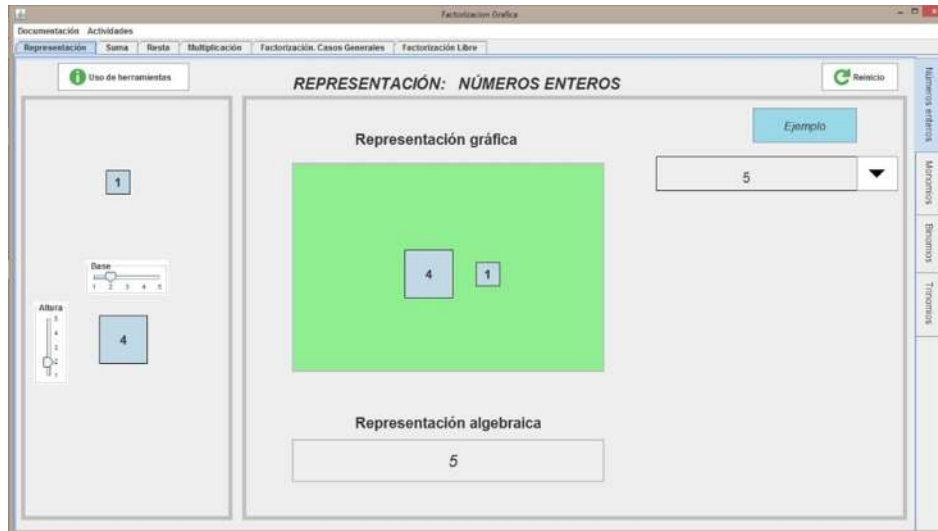


Figura 29. Ejemplo de la representación de un número entero.

Representación de polinomios:

Las secciones *representación de monomios*, *binomios* y *trinomios* son muy similares en la elección del ejercicio (ver Figura 30).



Figura 30. Sección para representar monomios.

En ellas el usuario deberá seleccionar el coeficiente y la parte literal de cada término del polinomio, según sea el caso, posteriormente deberá arrastrar los bloques necesarios al área de *Representación gráfica*, si la representación es correcta el recuadro donde se colocaron los bloques cambiará de color (ver Figura 31).

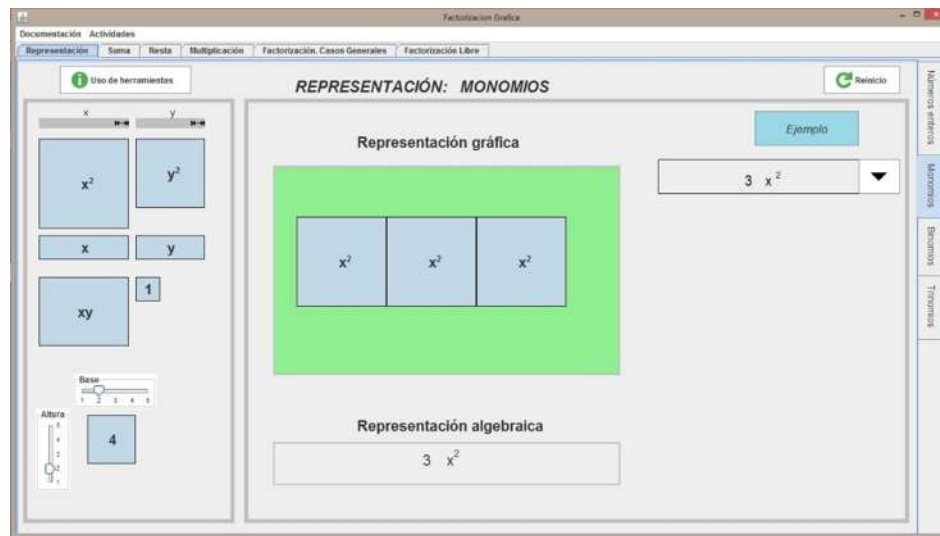


Figura 31. Ejemplo de la representación de un monomio.

SUMA DE POLINOMIOS

La pestaña de *suma* tiene cuatro secciones ubicadas en las pestañas laterales son: suma de números enteros, monomios, binomios y trinomios (ver Figura 32).

Suma de números enteros:

En esta parte al desplegar el menú para seleccionar un ejercicio aparecen dos paneles. El usuario deberá seleccionar dos números enteros, uno en cada panel (ver Figura 33), posteriormente deberá arrastrar los bloques que representen a cada número entero al área de Representación gráfica, si la suma es correcta el recuadro donde se colocaron los bloques cambiará de color.

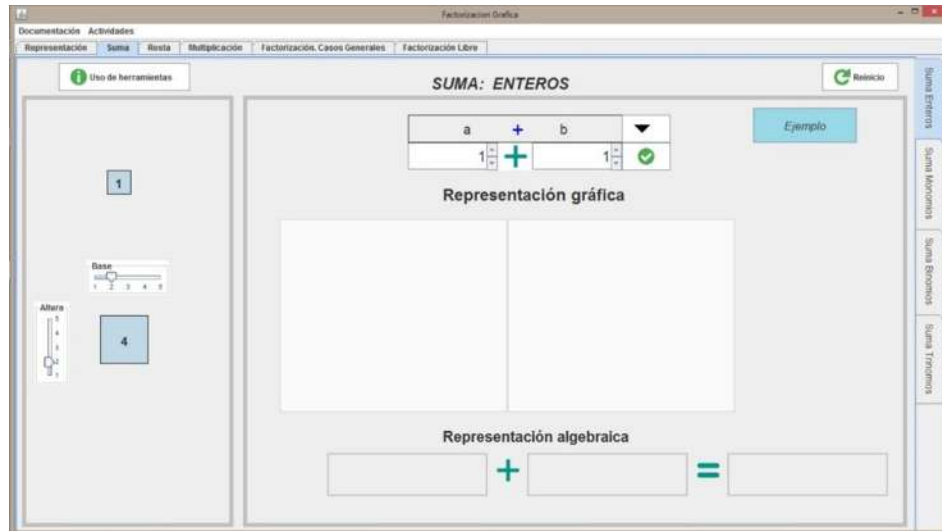


Figura 32. Sección para realizar sumas con números enteros.

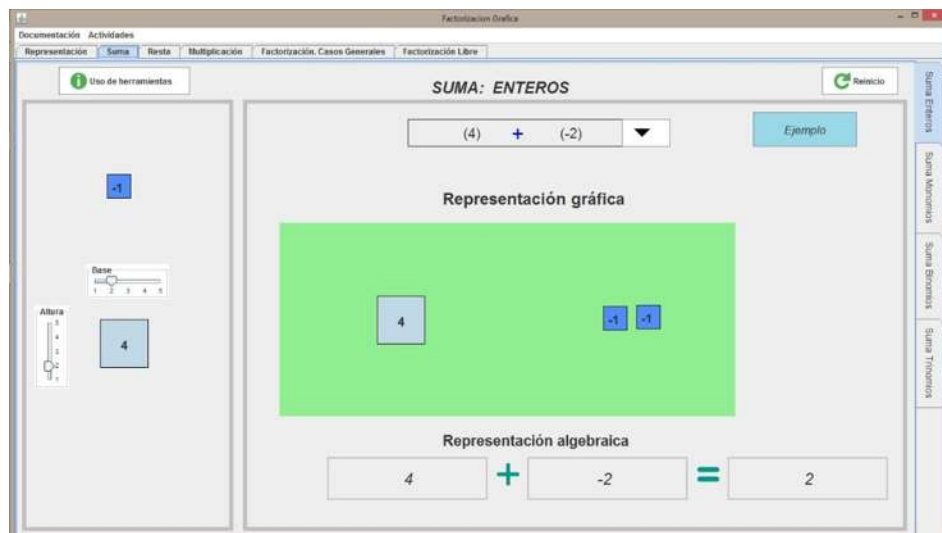


Figura 33. Ejemplo de la suma de dos números enteros.

Suma de polinomios:

En las secciones *suma de monomios*, *binomios* y *trinomios* se despliegan dos paneles para elegir dos polinomios (ver Figura 34). El usuario deberá seleccionar los coeficientes y las partes literales de cada término del polinomio, de cada sumando, según sea el caso, monomios, binomios o trinomios, posteriormente deberá arrastrar los bloques necesarios al área de *Representación gráfica*, si la suma es correcta el recuadro donde se colocaron los bloques cambiará de color (ver Figura 35).

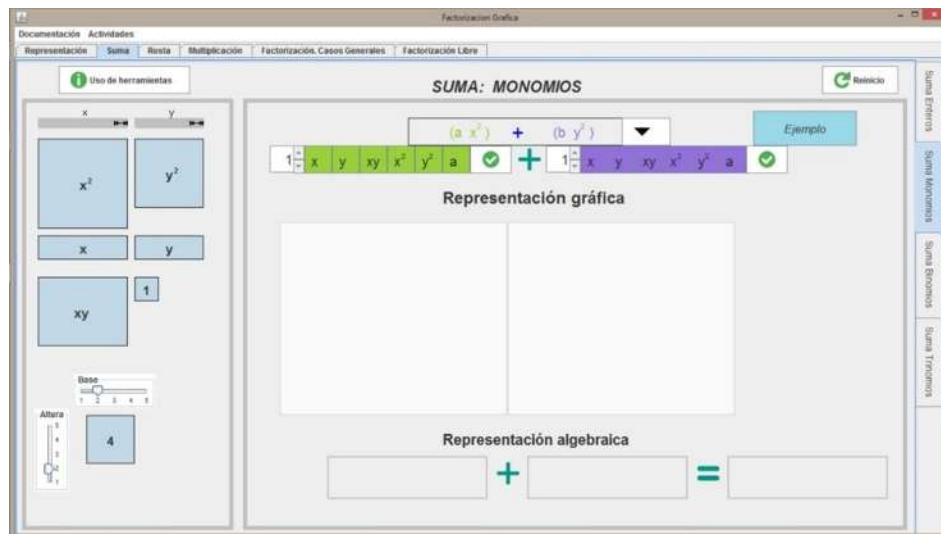


Figura 34. Sección para realizar sumas con monomios.

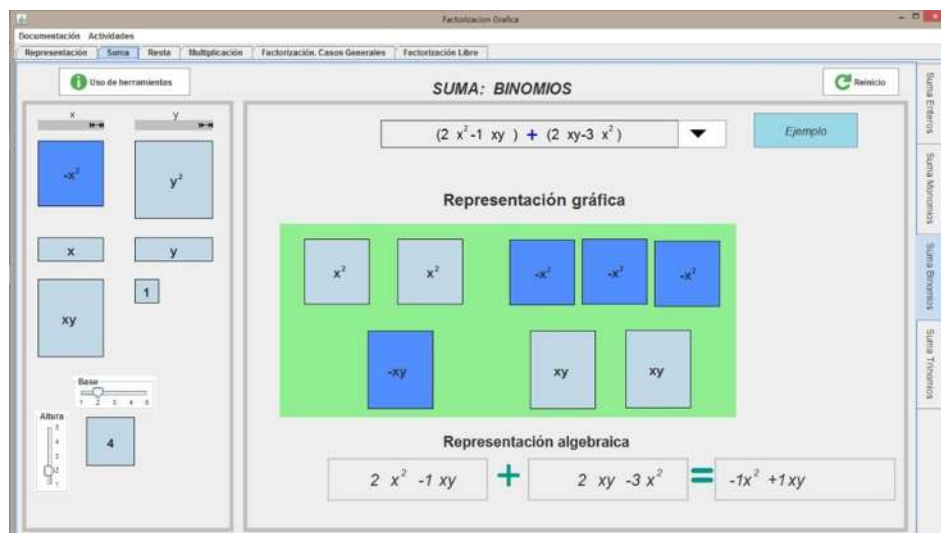


Figura 35. Ejemplo de la suma de dos binomios.

RESTA DE POLINOMIOS

La pestaña de *resta* tiene a su vez cuatro secciones ubicadas en las pestañas laterales, resta de números enteros, monomios, binomios y trinomios (ver Figura 36).

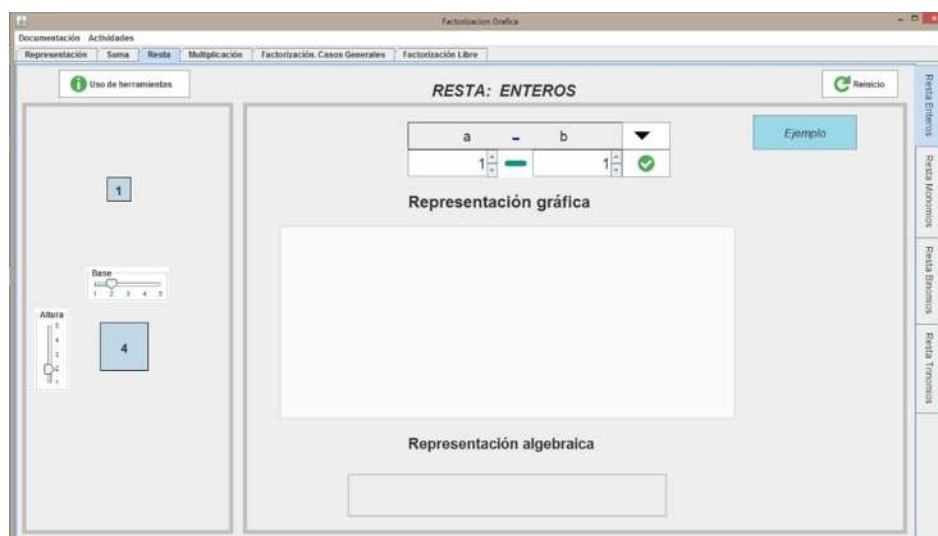


Figura 36. Sección para realizar restas con números enteros

Resta de números enteros:

En esta sección al desplegar el menú para seleccionar un ejercicio aparecen dos paneles. El usuario deberá seleccionar dos números enteros, posteriormente deberá arrastrar los bloques necesarios para representar el minuendo al área de *Representación gráfica*, después deberá quitar o restar los bloques del sustraendo, si el resultado de la resta es correcto, el recuadro donde se colocaron los bloques cambiará de color (ver Figura 37).

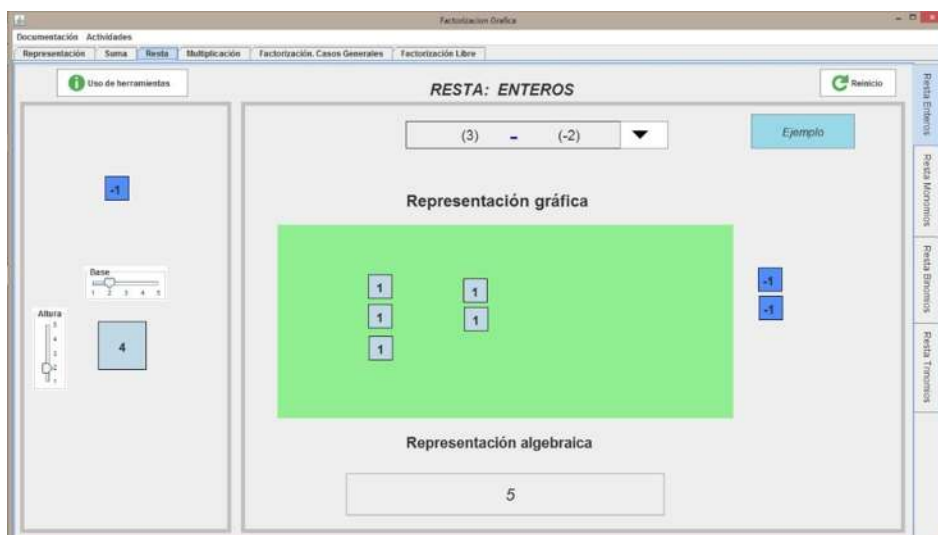


Figura 37. Ejemplo de la resta de dos números enteros.

Resta de polinomios:

En las secciones *resta de monomios*, *binomios* y *trinomios* se despliegan dos paneles al accionar el botón de “selecciona un ejercicio” (ver Figura 38).

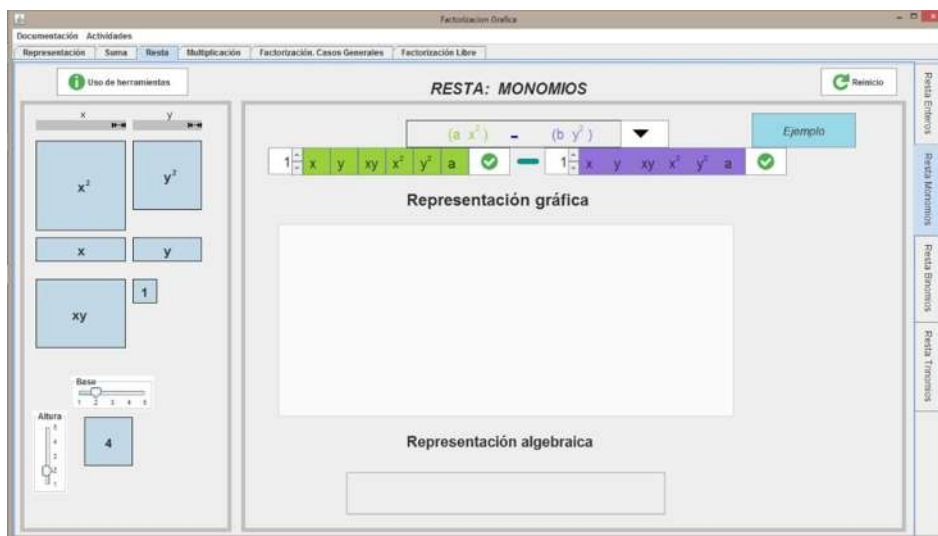


Figura 38. Sección para realizar restas con monomios.

El usuario deberá elegir el coeficiente y la literal de cada término de los polinomios, del minuendo y sustraendo, a continuación, deberá arrastrar los bloques necesarios del minuendo al área de *Representación gráfica*, después deberá quitar o restar los bloques del sustraendo, si la resta es correcta el recuadro donde se colocaron los bloques cambiará de color (ver Figura 39).

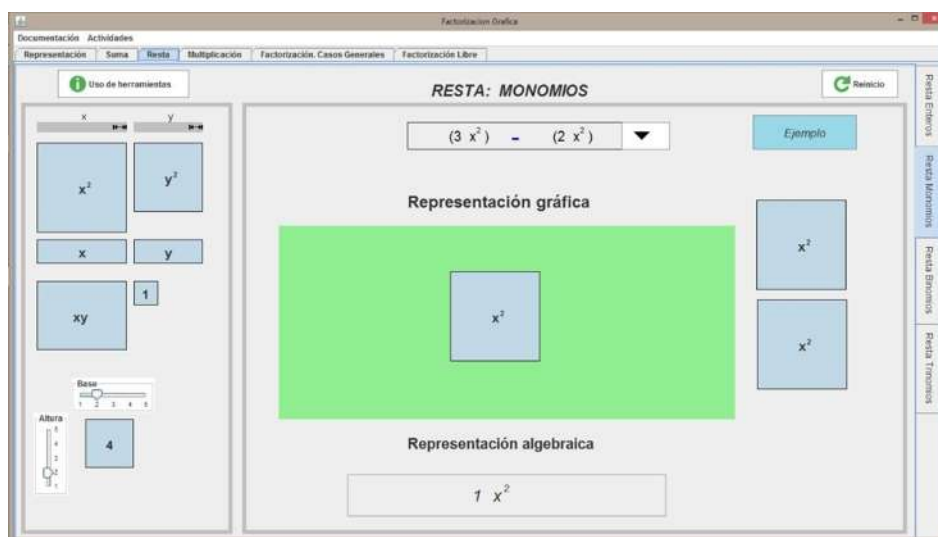


Figura 39. Ejemplo de la resta de dos monomios.

MULTIPLICACIÓN DE POLINOMIOS

La pestaña de multiplicación contiene las secciones de: construcción de bloque, monomio por monomio, monomio por binomio, monomio por trinomio y binomio por binomio, en ellas el usuario elegirá la base y la altura para construir un rectángulo como muestra la Figura 40.

Construcción de los bloques:

En esta sección se construyen los bloques que conforman las herramientas, al accionar el botón flecha aparecen dos menús A (Altura) y B (Base). En cada menú se tienen las

opciones de X , $-X$, Y , $-Y$, 1 y -1 donde el usuario deberá elegir una opción para la altura y una opción para la base y presionar el botón ok, inmediatamente aparecerá en la parte central el contorno del bloque elegido, así como las etiquetas de los lados que seleccionó.

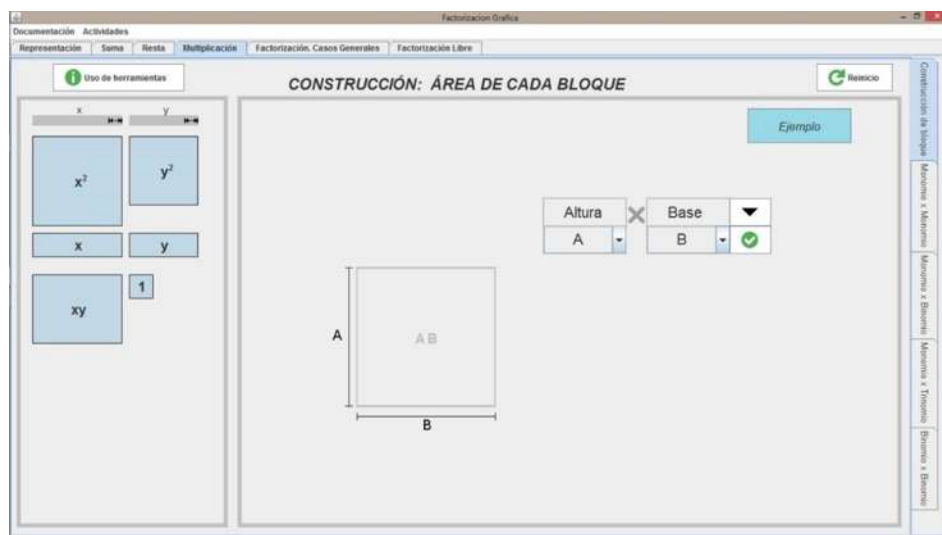


Figura 40. Sección para construir un bloque con diferente base y altura.

El usuario deberá colocar el bloque que tenga esa área dentro del contorno, si es el bloque correcto aparecerá el resultado del área y sus lados en color verde (ver Figura 41).

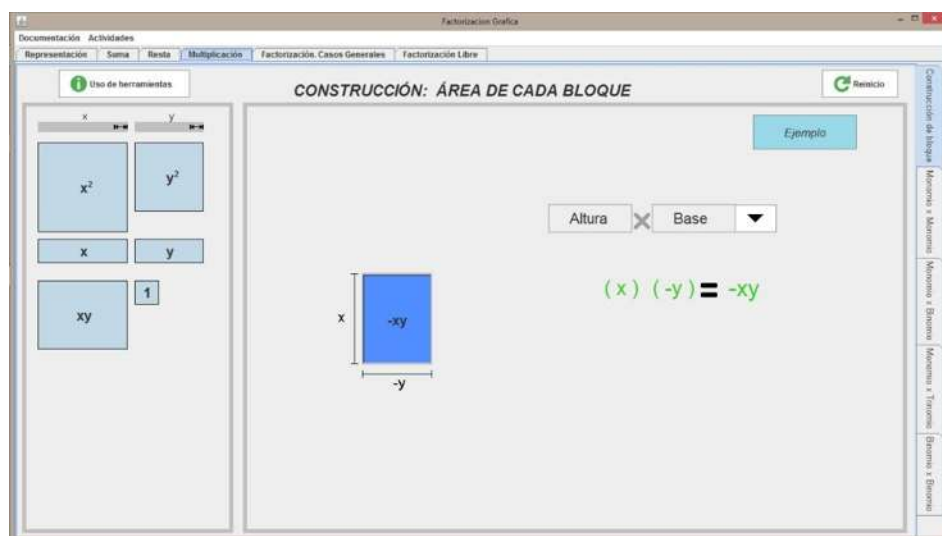


Figura 41. Ejemplo de la construcción del bloque $-xy$.

Monomio por Monomio

Al presionar el botón de flecha en esta sección, aparecerán dos paneles para seleccionar un monomio como la base y un monomio como altura de un rectángulo (ver Figura 42). Una vez elegido el coeficiente y la literal de los monomios, se deberá presionar el botón ok en cada panel y aparecerán las etiquetas de sus lados, así como del monomio resultante de la multiplicación de éstos en color negro.

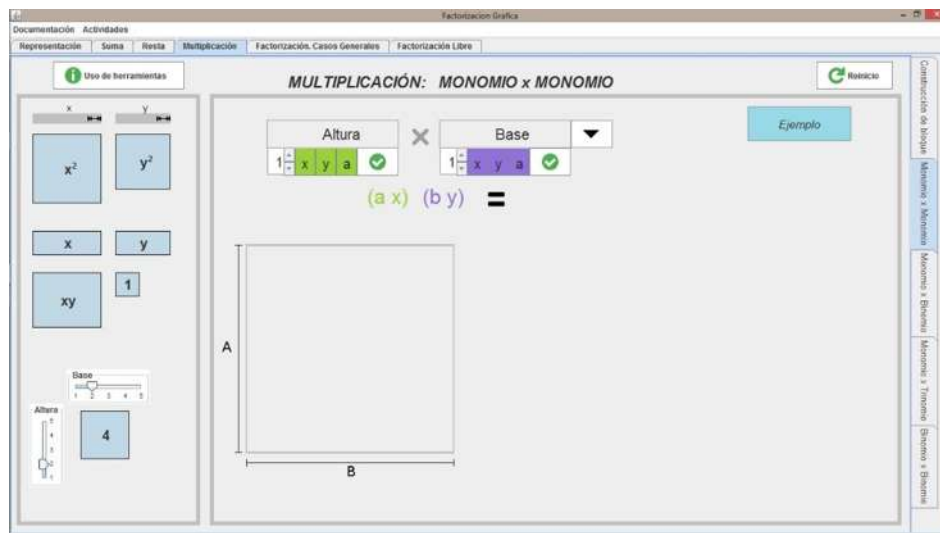


Figura 42. Sección para multiplicar monomio por monomio.

En la parte central de la aplicación aparecerá el contorno del rectángulo seleccionado y una etiqueta con el contador de los bloques que se encuentran dentro del contorno del rectángulo. El usuario deberá arrastrar y acomodar los bloques necesarios para cubrir el área del rectángulo, si la cantidad de bloques que cubren el área son correctos, los resultados cambiarán de color (ver Figura 43).

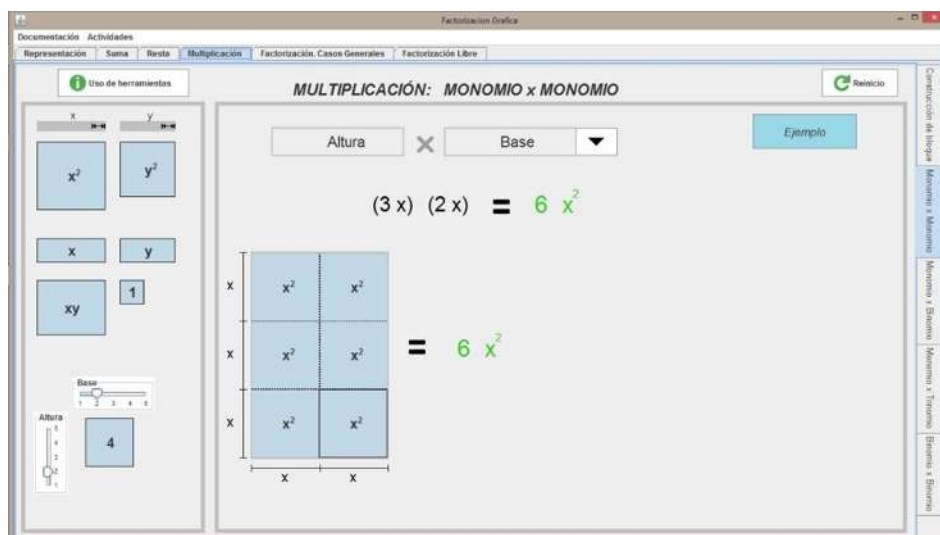


Figura 43. Representación de la multiplicación de un monomio por otro monomio.

Monomio por Binomio

Esta sección es similar a la anterior, al presionar el botón flecha aparecerán dos paneles para formar un rectángulo que tenga como altura un monomio y como base un binomio (ver Figura 44). Una vez elegido el coeficiente y la literal de los lados, se deberá presionar el botón ok en cada panel y aparecerán las etiquetas de sus lados, así como del binomio resultante de la multiplicación de éstos en color negro.

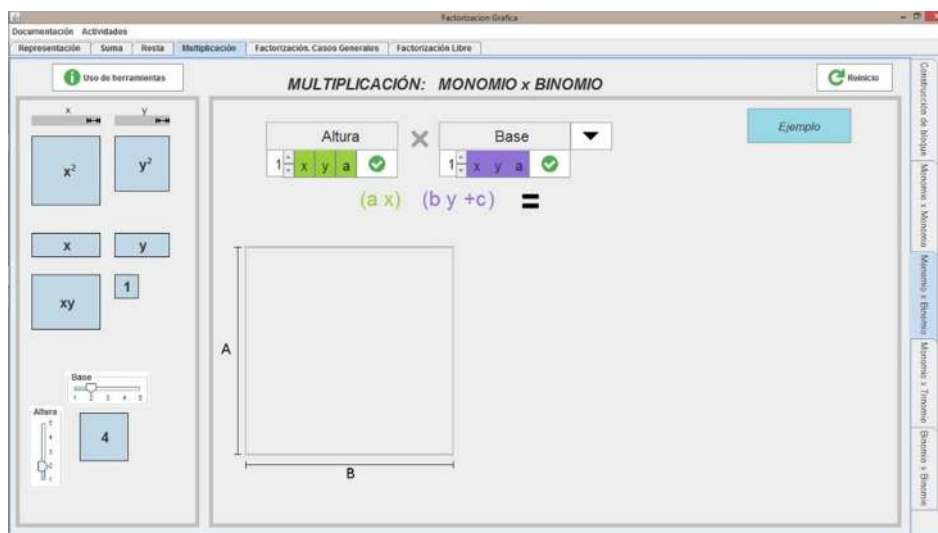


Figura 44. Sección para multiplicar monomio por un binomio.

En la parte central aparecerá el contorno del rectángulo y etiquetas con los contadores de los bloques que se encuentran dentro del contorno del rectángulo. El usuario deberá arrastrar y acomodar los bloques necesarios para cubrir el área del rectángulo, si la cantidad de bloques que cubren el área son correctos, los resultados cambiarán de color (ver Figura 45).

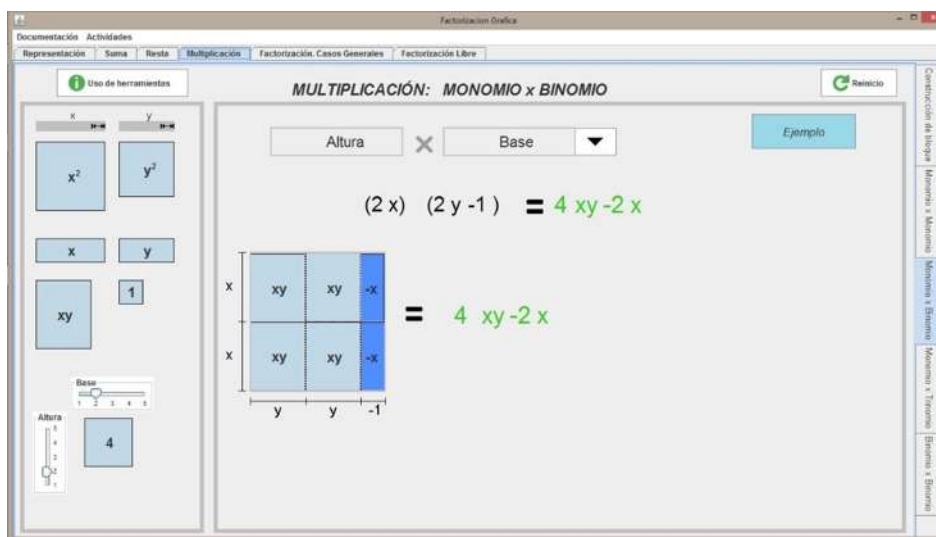


Figura 45. Representación de la multiplicación de un monomio por un binomio.

Monomio por Trinomio

En esta parte al presionar el botón flecha surgirán dos paneles para formar un rectángulo que tenga como altura un monomio y como base un trinomio (ver Figura 46). Una vez elegido el coeficiente y la literal de los lados, se deberá presionar el botón ok en cada panel y aparecerán las etiquetas de sus lados, así como del trinomio resultante de la multiplicación de éstos en color negro.

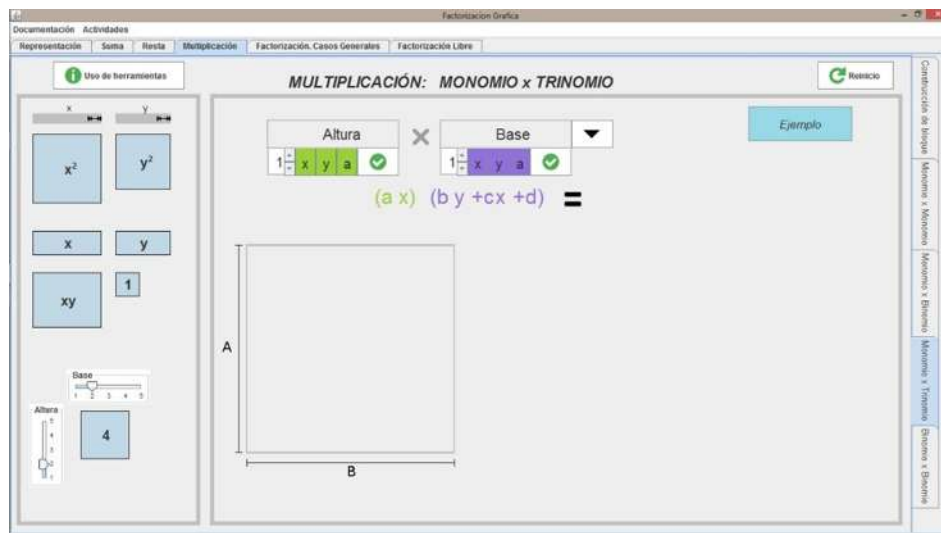


Figura 46. Sección para multiplicar monomio por un trinomio.

En la parte central aparecerá el contorno del rectángulo y etiquetas con los contadores de los bloques que se encuentran dentro del contorno del rectángulo. El usuario deberá arrastrar y acomodar los bloques necesarios para cubrir el área del rectángulo (ver Figura 47), si la cantidad de bloques que cubren el área son correctos, los resultados cambiarán de color.

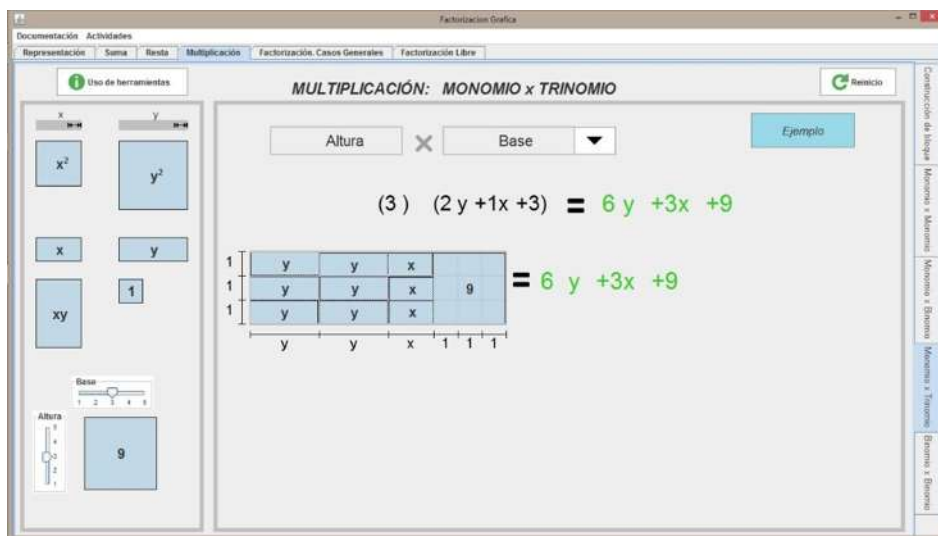


Figura 47. Representación de la multiplicación de un monomio por un trinomio.

Binomio por Binomio

En esta sección al oprimir el botón de flecha aparecerán dos paneles para seleccionar un binomio como la base y un binomio como altura de un rectángulo (ver Figura 48). Una vez elegido los coeficientes y las literales de los binomios, se deberá presionar el botón ok en cada panel y aparecerán las etiquetas de sus lados, así como del polinomio resultante de la multiplicación de éstos en color negro.

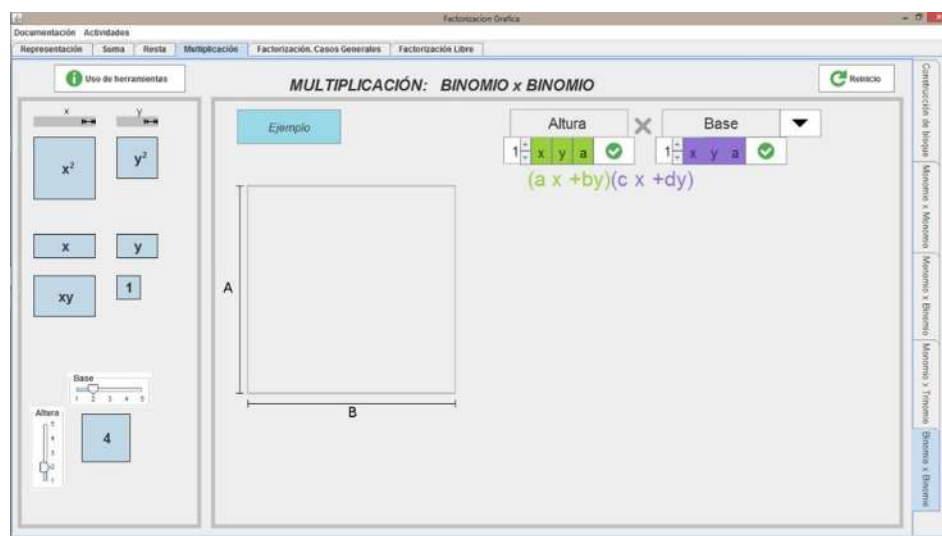


Figura 48. Sección para multiplicar un binomio por otro binomio.

En la parte central de la aplicación aparecerá el contorno del rectángulo seleccionado y una etiqueta con el contador de los bloques que se encuentran dentro del contorno del rectángulo. El usuario deberá arrastrar y acomodar los bloques necesarios para cubrir el área del rectángulo (ver Figura 49), si la cantidad de bloques que cubren el área son correctos, los resultados cambiarán de color.

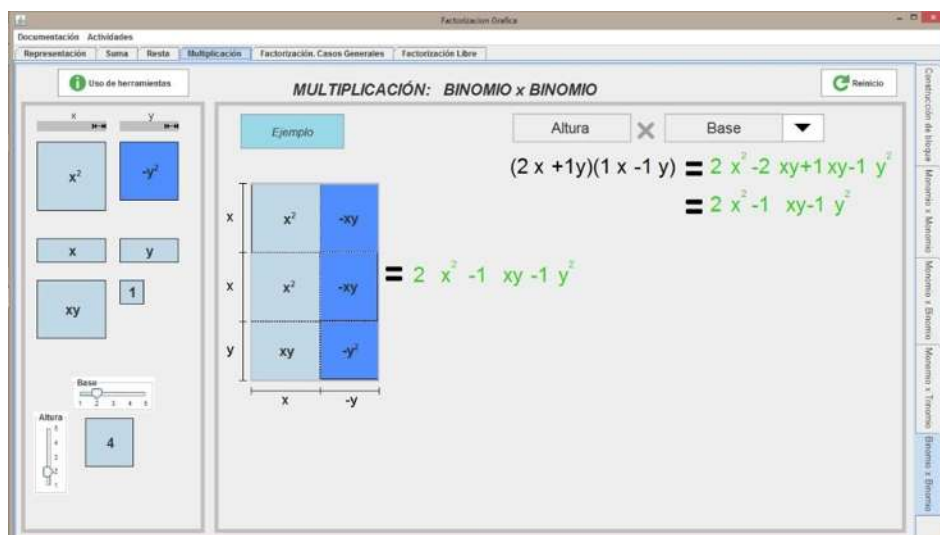


Figura 49. Representación de la multiplicación de un binomio por otro binomio.

FACTORIZACIÓN DE POLINOMIOS

En esta unidad se abordan los casos más comunes de factorización de polinomios, las secciones que contiene son: *factor común 1* para factorizar binomios por factor común, *factor común 2* para factorizar trinomios por factor común, *diferencia de cuadrados*, *trinomios 1* para factorizar trinomio cuadrado perfecto y *trinomios 2* para factorizar trinomios de la forma $x^2 + (a + b)x + ab$.

Factor común 1:

En esta sección el usuario al desplegar el panel, tendrá que elegir un binomio (ver Figura 50), si éste no tiene un factor común, aparecerá como mensaje “*Intenta con otro binomio*”. Si el binomio tiene factor común aparecerá el contorno de un rectángulo, así como sus lados y las etiquetas de los contadores de los bloques que se encuentran dentro de él, el usuario deberá arrastrar y acomodar los bloques necesarios para cubrir el rectángulo que representa el binomio, si es correcta los contadores cambiarán de color (ver Figura 51).

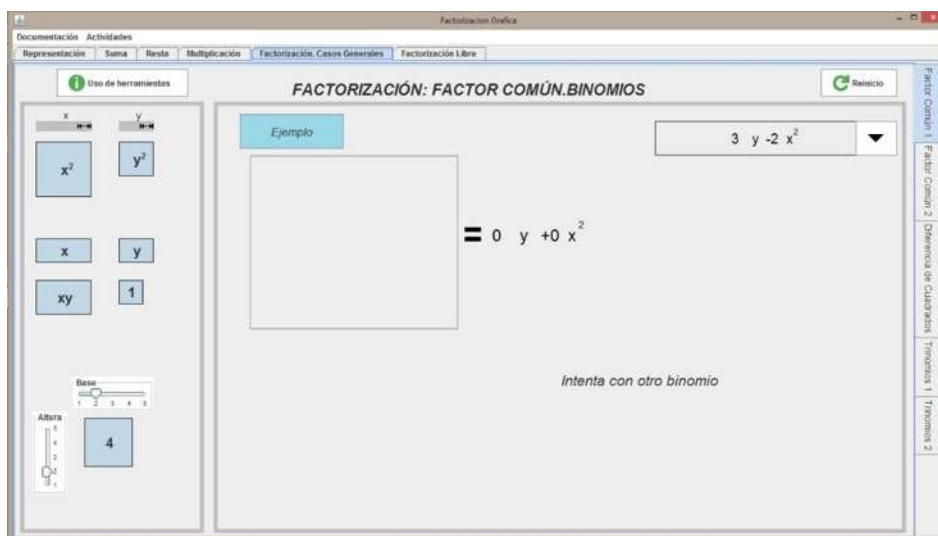


Figura 50. Sección para factorizar un binomio por factor común.

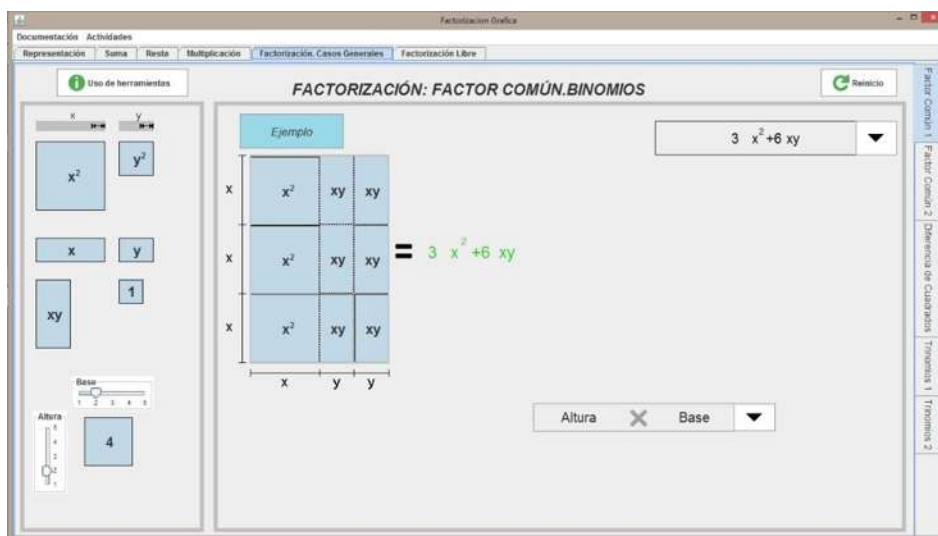


Figura 51. Representación de un binomio con factor común.

También aparecerá en la parte inferior derecha dos paneles en los cuales el usuario deberá seleccionar la base y la altura del rectángulo (ver Figura 52). Si la elección es correcta las etiquetas con la base, altura y el resultado cambiarán de color.

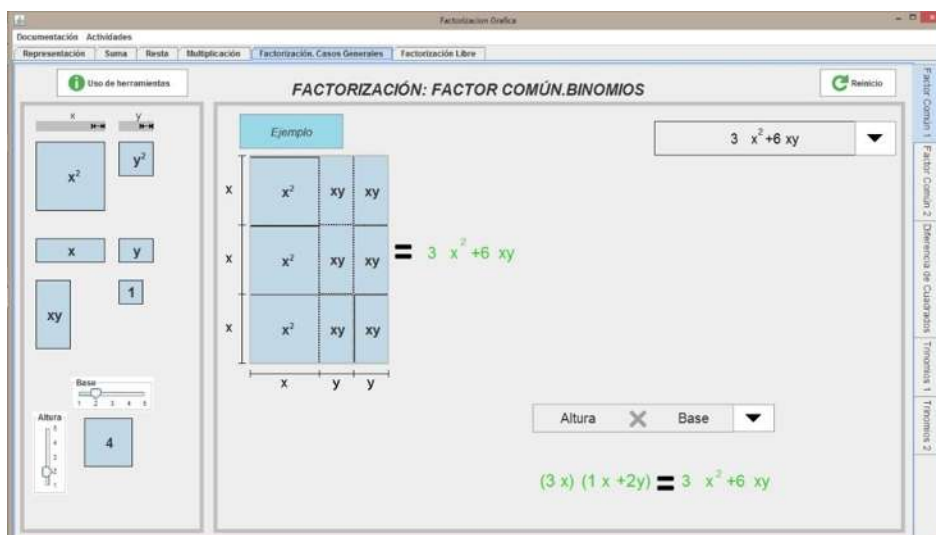


Figura 52. Factorización de un binomio con factor común.

Factor común 2:

En esta sección el usuario al desplegar el panel, tendrá que elegir un trinomio, si éste no tiene un factor común, aparecerá como mensaje “*Intenta con otro trinomio*” (ver Figura 53).

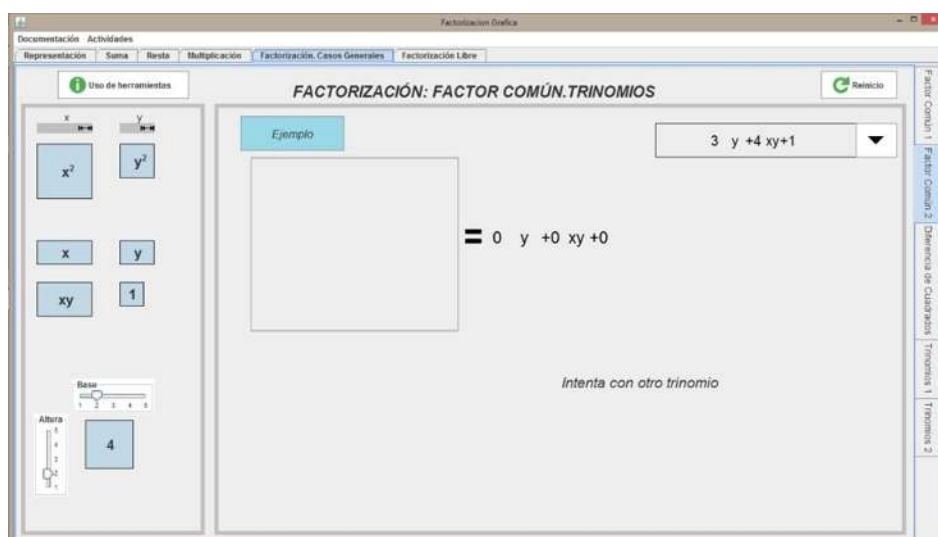


Figura 53. Sección para factorizar un trinomio por factor común.

Si el trinomio tiene factor común aparecerá el contorno de un rectángulo, así como sus lados y las etiquetas de los contadores de los bloques que se encuentran dentro de él, el usuario deberá arrastrar y acomodar los bloques necesarios para cubrir el rectángulo que representa el trinomio, si es correcta los contadores cambiarán de color (ver Figura 54).

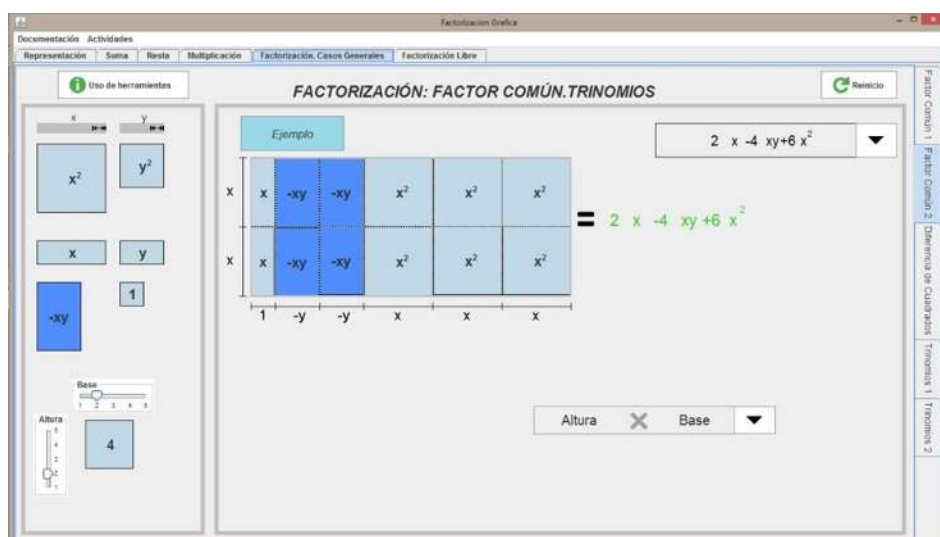


Figura 54. Representación de un trinomio con factor común.

También aparecerá en la parte inferior derecha dos paneles en los cuales el usuario deberá seleccionar la base y la altura del rectángulo. Si la elección es correcta las etiquetas con la base, altura y el resultado cambiarán de color (ver Figura 55).

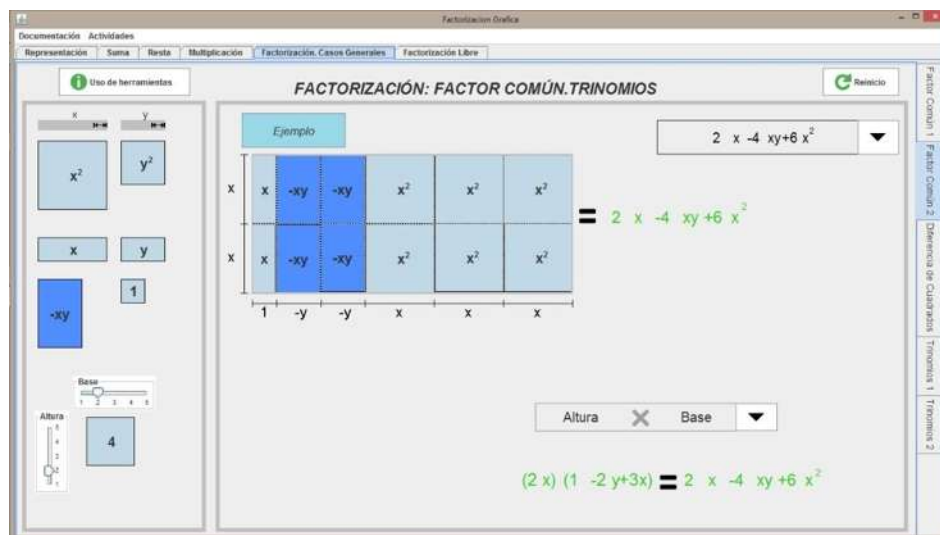


Figura 55. Factorización de un trinomio con factor común.

Diferencia de cuadrados:

Al desplegar el panel el usuario tendrá que elegir un binomio que sea una diferencia de cuadrados (ver Figura 56), si no es el caso aparecerá el siguiente mensaje “*Intenta con otro binomio*”.

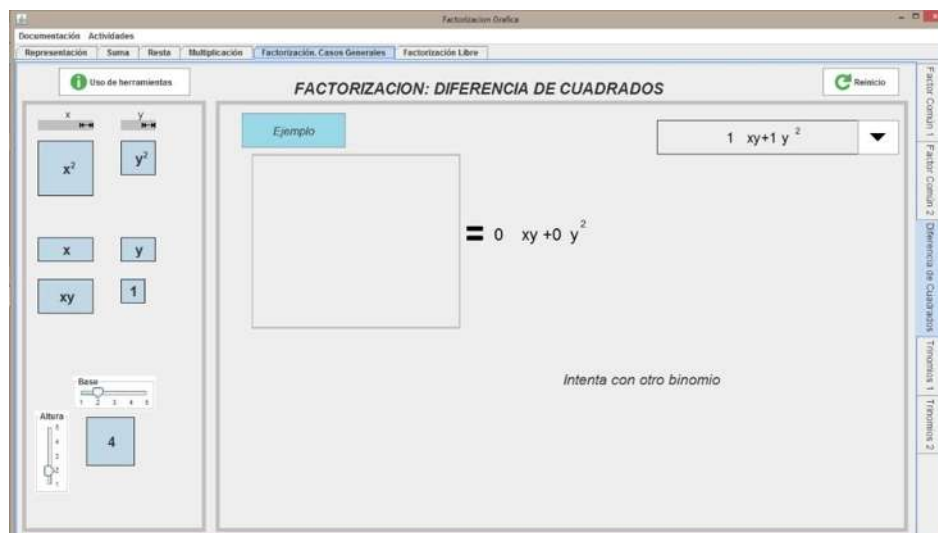


Figura 56. Sección para factorizar un binomio que sea una diferencia de cuadrados.

Si la elección del binomio es una diferencia de cuadrados aparecerá el contorno de un rectángulo, así como sus lados y las etiquetas de los contadores de los bloques que se encuentran dentro de él, además de las etiquetas del binomio aparece un término más el cual permite que se realice la factorización. El usuario deberá arrastrar y acomodar los bloques necesarios para cubrir el rectángulo que representa el trinomio, si es correcta los contadores cambiarán de color (ver Figura 57).



Figura 57. Representación de un binomio que es una diferencia de cuadrados.

Así mismo, aparecerá en la parte inferior derecha dos paneles en los cuales el usuario deberá seleccionar la base y la altura del rectángulo. Si la elección es correcta las etiquetas con la base, altura y el resultado cambiarán de color (ver Figura 58).

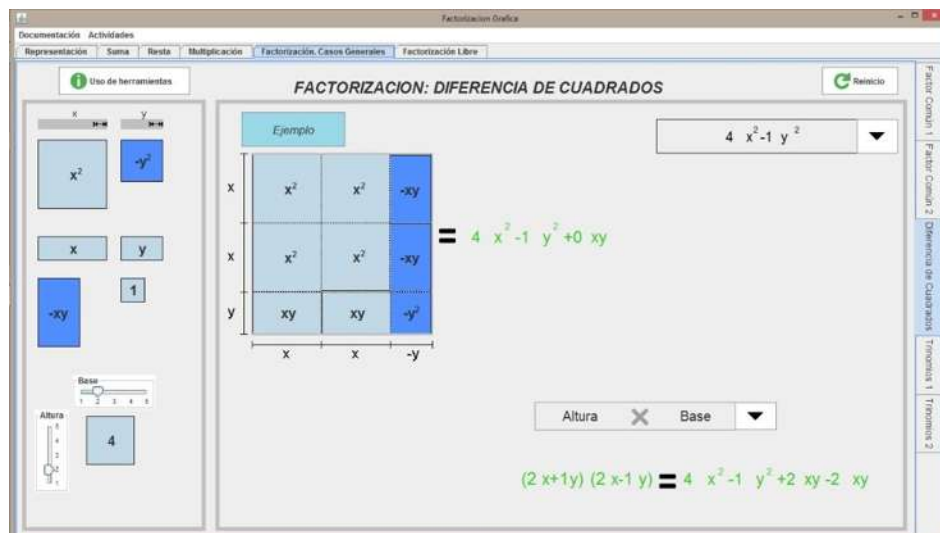


Figura 58. Factorización de un binomio que es una diferencia de cuadrados.

Trinomios 1:

En esta parte el usuario tendrá que elegir un trinomio cuadrado perfecto, si no lo es aparecerá el siguiente mensaje “*Intenta con otro trinomio*” (ver Figura 59).

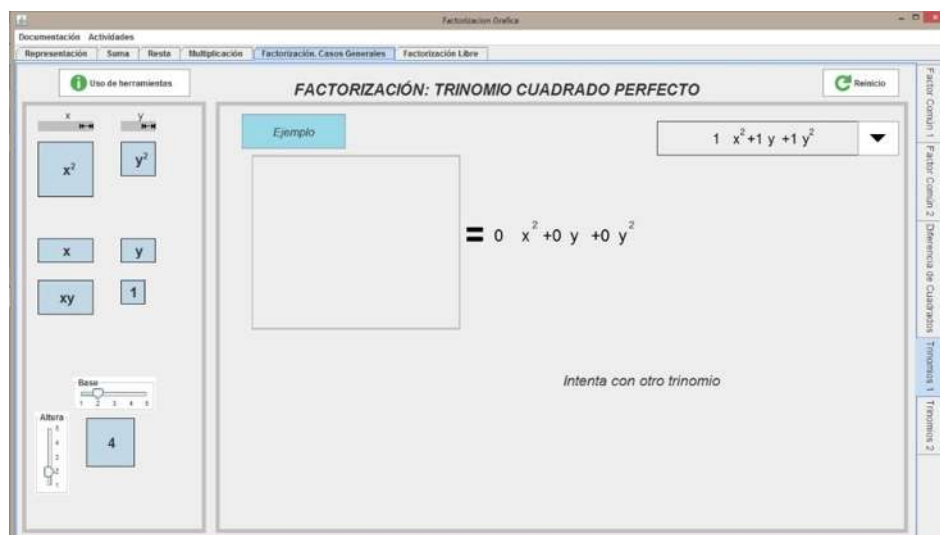


Figura 59. Sección para factorizar un trinomio cuadrado perfecto.

Si es un trinomio cuadrado perfecto aparecerá el contorno de un rectángulo, así como sus lados y las etiquetas de los contadores de los bloques que se encuentran dentro de él, el usuario deberá colocar los bloques necesarios para cubrir el rectángulo que representa el trinomio, si es correcto los contadores cambiarán de color (ver Figura 60).

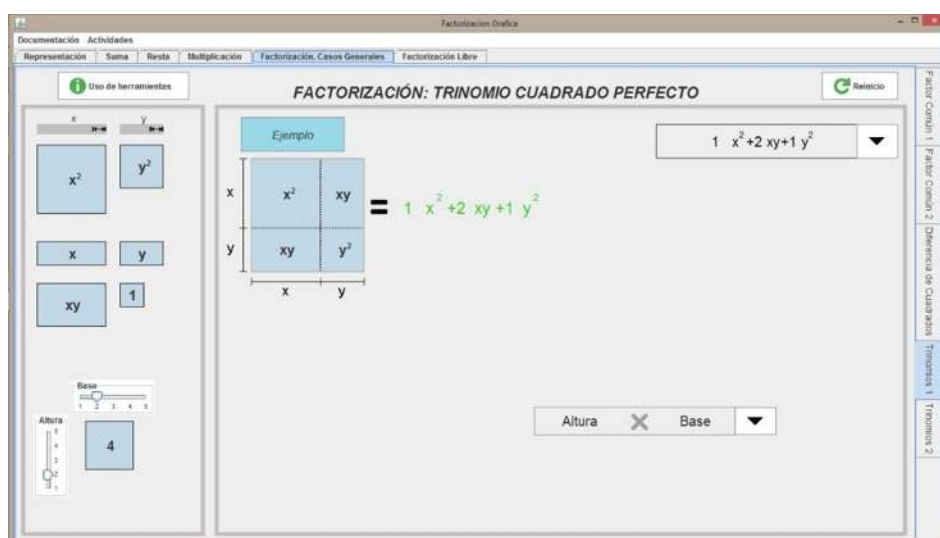


Figura 60. Representación de un trinomio cuadrado perfecto.

Del mismo modo aparecerá en la parte inferior derecha dos paneles en los cuales el usuario deberá seleccionar la base y la altura del rectángulo. Si es correcta la elección cambiarán de color las etiquetas con la base, altura y el resultado (ver Figura 61).

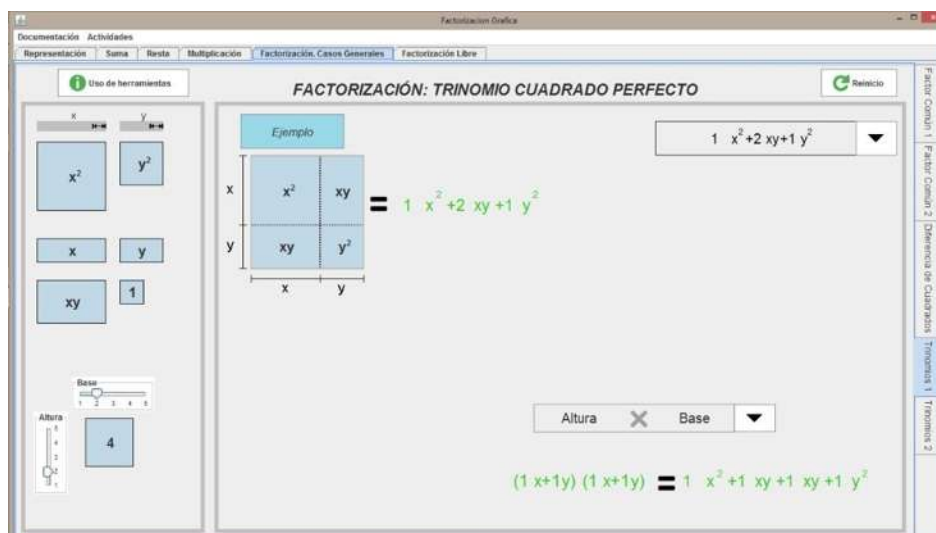


Figura 61. Factorización de un trinomio cuadrado perfecto.

Trinomios 2:

En esta sección el usuario tendrá que elegir un trinomio de la forma $x^2 + (a + b)x + ab$, si no lo es aparecerá el siguiente mensaje “Intenta con otro trinomio” (ver Figura 62).

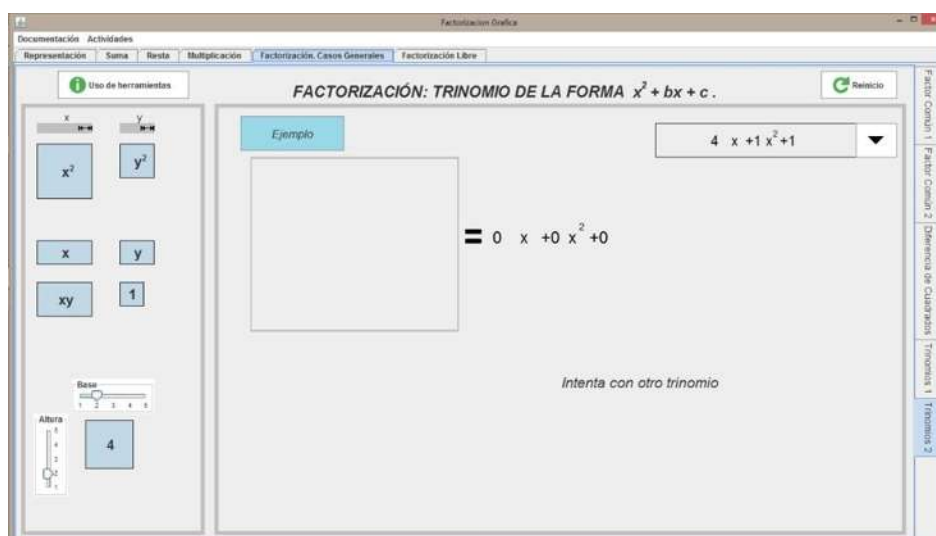


Figura 62. Sección para factorizar un trinomio cuadrado no perfecto.

Si cumple con la forma antes mencionada aparecerá el contorno de un rectángulo, así como sus lados y las etiquetas de los contadores de los bloques que se encuentran dentro de él, el usuario deberá colocar los bloques necesarios para cubrir el rectángulo que representa el trinomio, si es correcto los contadores cambiarán de color (ver Figura 63).

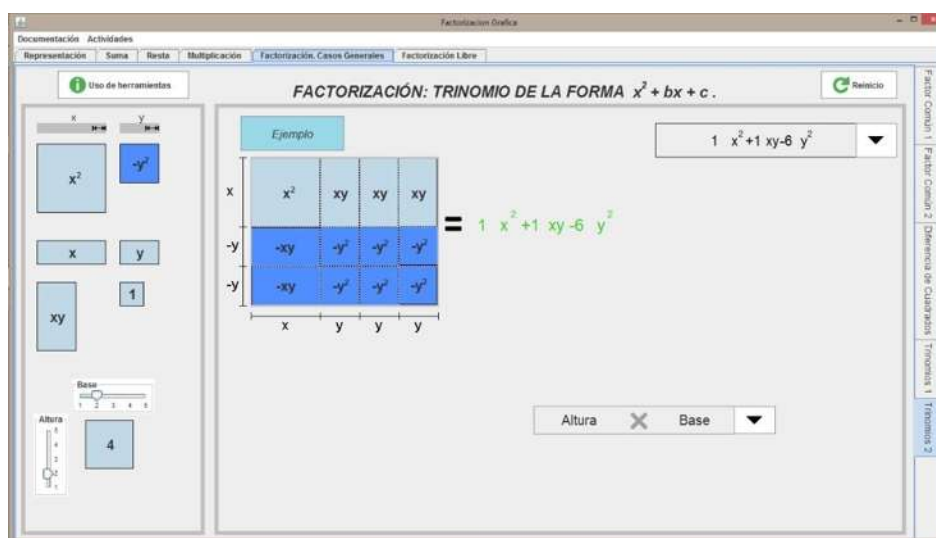


Figura 63. Representación de un trinomio cuadrado no perfecto.

También en la parte inferior derecha emergerán dos paneles en los cuales se deberá seleccionar la base y la altura del rectángulo. Si es correcta la elección cambiarán de color las etiquetas con la base, altura y el resultado (ver Figura 64).

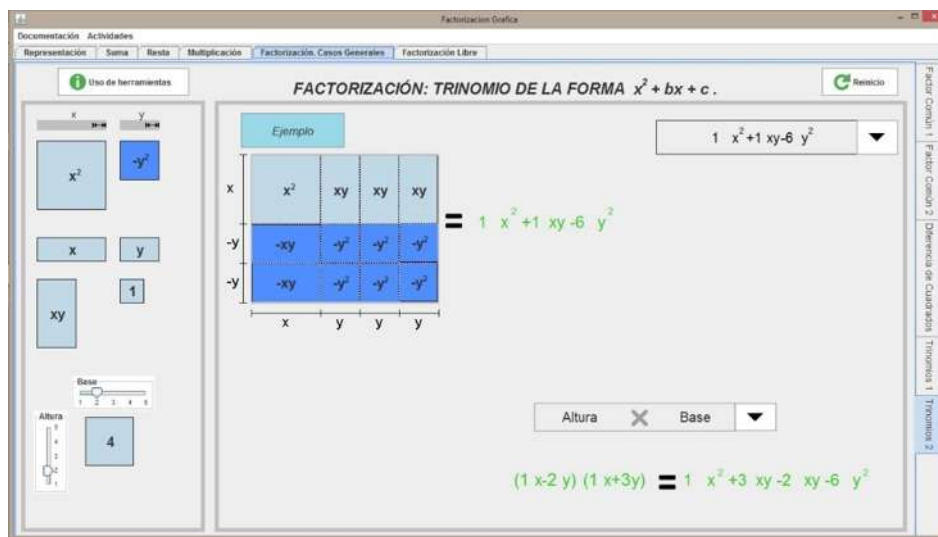


Figura 64. Factorización de un trinomio cuadrado no perfecto.

FACTORIZACIÓN LIBRE DE POLINOMIOS

En esta unidad se muestra en la sección de herramientas los siguientes elementos adicionales: contadores de cada bloque, un símbolo de igual y pequeños rectángulos para colocar en la base y altura del rectángulo (ver Figura 65), los cuales también cambian de signo y orientación y se pueden mover al área central, en la sección central se encuentra un área blanca la cual puede ajustar su tamaño moviendo el icono con cuatro flechas, en ella se deberán arrastrar y acomodar los bloques.

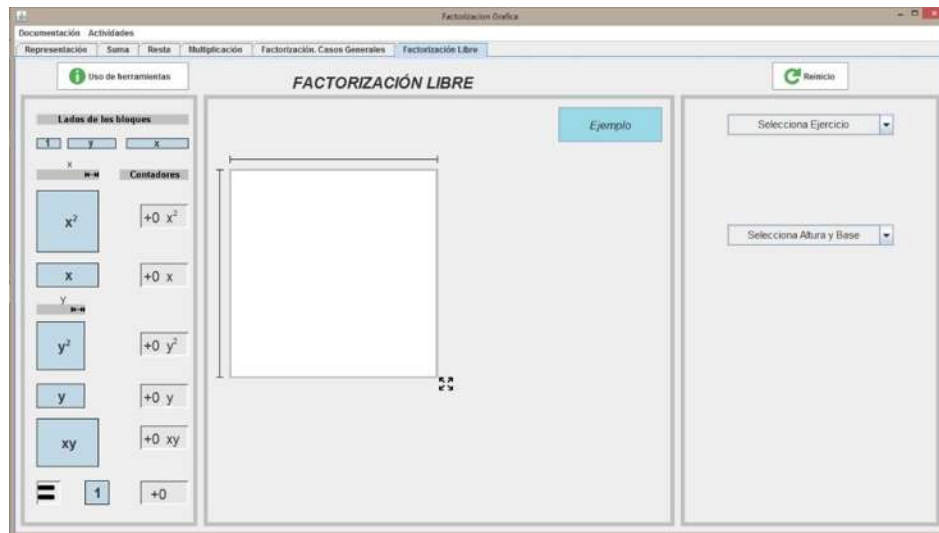


Figura 65. Sección para representar y factorizar libremente algún polinomio.

En la sección derecha se encuentran dos menús (ver Figura 66), en el superior se podrá elegir el tipo de polinomio que se quiere factorizar (monomio, binomio, trinomio o cuatrinomio), posteriormente deberán seleccionar los coeficientes y literales, según sea el caso. Una vez elegido el ejercicio y presionado el botón *ok*, aparecerá el polinomio en color negro.

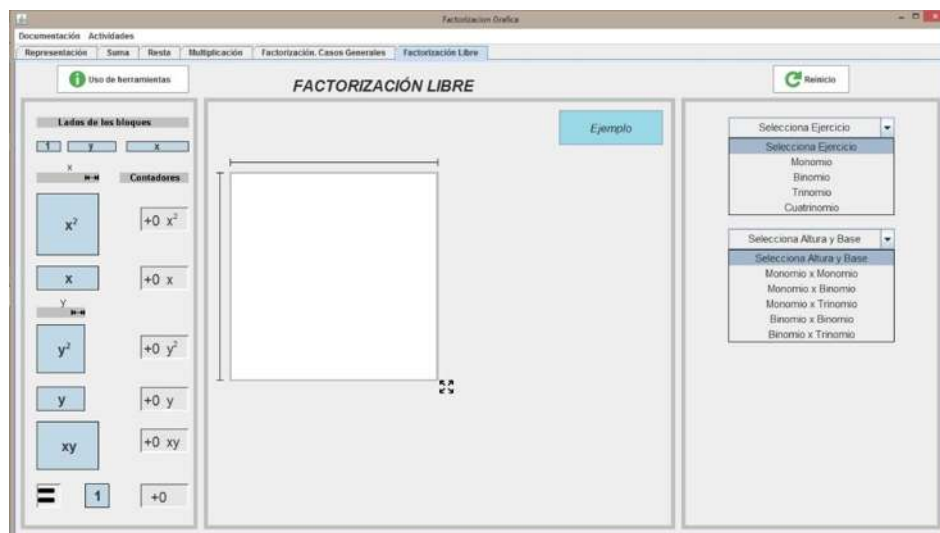


Figura 66. Despliegue de paneles para elegir un polinomio, así como la base y altura del mismo.

En el menú inferior el usuario podrá elegir la base y la altura, una vez que el usuario realizó su selección y presiono los botones ok, se desplegará el desarrollo de la multiplicación de la base por altura elegidos (ver Figura 67).

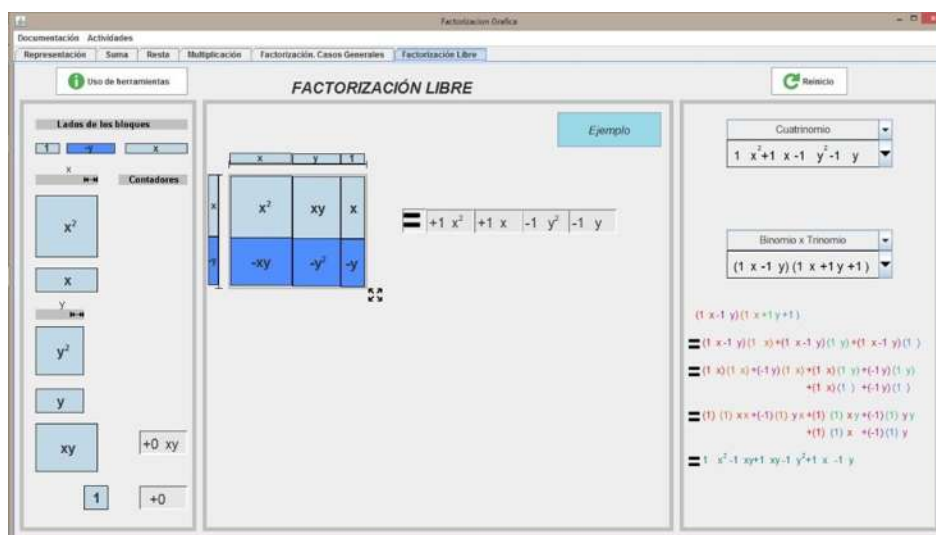


Figura 67. Representación de un cuatrinomio y su factorización.

En esta sección se da al usuario libertad de elegir y jugar con diferentes tipos de polinomios que no se incluyen en la sección de Factorización de casos generales, ya que puede acomodar a su gusto los bloques visualizando de manera gráfica y algebraica los términos que usa.

IV.4. ANÁLISIS DEL CÓDIGO

La clase *PruebaContenedor*

La clase *PruebaContenedor* inicia y asigna la posición y dimensiones de la clase *ContenedorGeneral* la cual contiene todos los elementos del programa.

La clase *ContenedorGeneral*

La clase *ContenedorGeneral* abre un *JFrame* donde se crean los elementos principales de la interfaz gráfica. Se crea un *JMenuBar* el cual es una barra o menú superior que contiene las opciones de Documentación y Actividades creadas con *JMenu*.

Dentro de cada opción de Documentación y Actividades se encuentras otras opciones relacionadas con las actividades de cada sección del programa, éstas son creadas con elementos *JMenuItem*.

```
JMenuBar MenuBar=new JMenuBar();  
JMenu menuPrincipal=new JMenu(String nombreMenu);  
JMenuItem SubMenu=new JMenuItem(String nombreSubMenu);
```

Se crea un elemento de tipo *JTabbedPane* el cual agrega cada sección del programa mediante pestañas horizontales en la parte superior del programa.

```
JTabbedPane pestanas=new JTabbedPane();  
pestanas.addTab(String nombrePestaña, Component seccion);
```

Cada una de las pestañas creadas contienen las siguientes secciones del programa: Representación, Suma, Resta, Multiplicación, Factorización Casos Generales y Factorización Libre. En cada sección se crea también un elemento de tipo *JTabbedPane* el cual agrega cada subsección mediante pestañas verticales en el costado derecho de la aplicación.

CLASES GENERALES

a) La clase Creado Complementos

Esta clase, crea gráficos para mejorar la apariencia de la interfaz. Sus métodos en general son:

Pintar_Objeto(String nombre, int xi ,int yi, float ancho, float largo, Graphics g);

Recibe como parámetros el nombre, la posición inicial del objeto, así como el ancho y largo de la figura, se hace uso de la clase **Graphics** y de algunos de sus métodos.

b) La clase CreadorTutorialGalería

La clase *CreadorTutorialGalería* abre un JFrame donde se muestra una galería fotográfica sobre el uso de las herramientas o un ejercicio de ejemplo.

CreadorTutorialGaleria galería=new CreadorTutorialGaleria(String Titulo, ImageIcon Img_Tutorial[]);

Recibe como parámetros una cadena de texto para el título de la ventana y un arreglo de imágenes que se mostrarán.

c) La clase Creador Lista Polinomios

Esta clase crea pequeños paneles donde se encuentran los botones para elegir el coeficiente y la variable de los polinomios. Sus métodos en general usan la siguiente estructura:

set_PanelBotones (int c)

El parámetro entero c establece el rango desde de $-c$ a $+c$ que tomarán las opciones para elegir el coeficiente de cada ejercicio.

d) La clase Encontrar Factores

La clase Encontrar Factores es usada en la sección de Factorización.

Para factorizar *binomios por factor común* se usan los métodos:

setAB(int a, int b) y get_MCD_AB()

El primer método recibe los coeficientes enteros de los términos del binomio y el segundo regresa el factor común de estos dos enteros si es que existe.

Para factorizar *trinomios por factor común* se usan los métodos:

setABC(int a, int b, int c) y get_MCD_ABC()

El primer método recibe los coeficientes enteros de los términos del trinomio y el segundo regresa el factor común de estos tres enteros si es que existe.

Para factorizar *binomios que son diferencia de cuadrados* se usan los métodos:

setAB_Raiz (int a, int b), getAA_Raiz (), getBB_Raiz() y getBoolean_Raices()

El primer método recibe los coeficientes enteros de los términos del binomio, el segundo y el tercero regresan las raíces de los coeficientes recibidos y el cuarto método regresa verdadero o falso dependiendo de si se encontraron raíces o no.

Para factorizar *trinomio cuadrado perfecto* se usan los métodos:

setAC_RaizTrin1(int a, int c), getAA_RaizTrin1(), getCC_RaizTrin1() y getBoolean_RaicesTrin1()

El primer método recibe el primer y tercer coeficiente entero de los términos del trinomio, el segundo y el tercer método regresan las raíces de los coeficientes recibidos y el cuarto método regresa verdadero o falso dependiendo de si se encontraron raíces o no.

Para factorizar *trinomio cuadrado no perfecto* se usan los métodos:

setAC_RaizTrin2(int a, int b, int c), getAA_RaizTrin2(), getCC_RaizTrin2() y getBoolean_RaicesTrin2()

El primer método recibe los coeficientes enteros de los términos del trinomio, el segundo y el tercer método regresan las raíces de los coeficientes recibidos y el cuarto método regresa verdadero o falso dependiendo de si se encontraron raíces o no.

LOS MÉTODOS PRINCIPALES

Los métodos principales son usados en cada sección del programa. En cada sección se crean 10 o más bloques de cada uno x^2 , x , y^2 , y , xy , 1 , ab (ver figura 68), la cantidad de bloques es guardada en la variable *numBloques*.

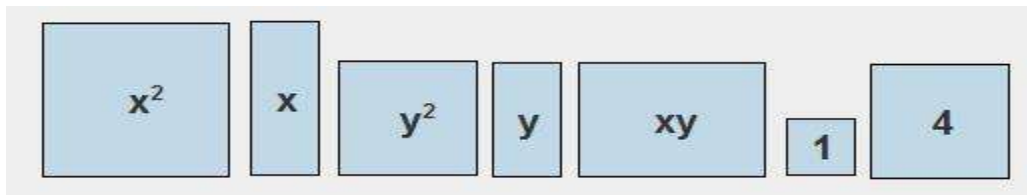


Figura 68. Representación de los bloques que contiene las herramientas.

Los bloques tienen las siguientes variables asociadas almacenadas en arreglos:

```
JPanel Bloques []=new JPanel[int numBloques]
boolean mover[]=new boolean[int numBloques]
boolean soltar[]=new boolean [int numBloques]
boolean arrastrar[]=new boolean[int numBloques]
int Contador[]=new int[int numBloques]
int tamaño_x[]=new int[int numBloques]
int tamaño_y[]=new int[int numBloques]
int Posición[][]=new int[int numBloques][2]
JLabel nombre[]=new JLabel[int numBloques]
```

a) Método CreadorBloques

Este método inicializa las variables de todos los bloques, creando, posicionando y asignándoles un color inicial de la siguiente manera:

```
CreadorBloques(){  
    ...  
    JPanel Bloque=new JPanel();  
    JPanel Bloque []=new JPanel[numBloques]  
    Bloque[].setBounds(Posicion[][0],           Posicion[][1],  
tamaño_x[],tamaño_y[]);  
    Bloque [].setBackground(Color);  
    ...  
}
```

b) Método actionPerformed

Este método indica las acciones que se deben realizar al interactuar con los botones herramientas, reinicio y ejemplo de la aplicación, así como con los botones creados con la clase *creadorListaPolinomios*, las elecciones del usuario se guardan en variables de tipo JLabel, String e int para los coeficientes. Al presionar el botón de Herramientas o el botón de ejemplo se crea y visualiza un JFrame creado con la clase Creador Tutorial Galería, al presionar el botón de reinicio se reinician todas las variables a la configuración inicial.

```
actionPerformed ( ActionEvent a ){  
    ...  
    If (a.getSource()==botón ){  
    ...  
    }  
}
```

c) Método **activacionBloques**

Cuando el usuario selecciona un ejercicio mediante los botones de la interfaz, los datos son guardados en las variables *seleccionV* (selección de la variable) y *seleccionE* (grado asociado a la variable), este método las compara con posibles elecciones y activa los contadores de la expresión seleccionada, el siguiente es un ejemplo de comparación con la expresión x^2 .

```
activacionBloques(){  
    ...  
    If ("x".equals( seleccionV ) && "2".equals( seleccionE )){  
        contador_C1 = contadorBloques_XX;  
    }  
    ...  
}
```

Las variables de tipo contador guardan la cuenta de los bloques que se sueltan o se quitan de cierta área de la aplicación.

d) Método **reinicio**

Se activa al presionar el botón de reinicio o al seleccionar un nuevo ejercicio. Este método regresa todas las variables a su configuración inicial, tanto en color, posición y valor.

```
reinicio(){  
    ...  
}
```

e) Método **estaDentro**

Este método revisa si el cursor se encuentra dentro o fuera de cierto objeto, ya sea área o bloque.

```

estaDentro_Objeto ( MouseEvent e ){
    return((cursorX > x)&&(cursorX < x + ancho)&&(cursorY > y
)&&(cursorY < y + largo));
}

```

Los métodos **mouseDragged**, **mouseMoved**, **mouseClicked** y **mouseReleased** son activados con los eventos del ratón. **MouseDragged** se llama una vez cada vez que se mueve el cursor mientras se pulsa un botón del ratón. Si no se pulsa un botón, **mouseMoved** () se llama en su lugar. **MouseClicked** () se llama después de que un botón del ratón ha sido presionado y luego soltado, y **mouseReleased** () se llama cada vez que se suelta el botón del ratón.

f) Método **mouseDragged**

Este método permite que se puedan arrastrar los bloques cambiando su posición, pero conservando sus características y forma. Cada bloque es revisado mediante una sentencia similar. Si algún bloque se está moviendo el valor de las variables **arrastrar []** de todos los objetos que no lo están se inhabilitan para que solo se pueda arrastrar un objeto a la vez.

```

mouseDragged ( MouseEvent e ){

    if(estaDentro_Objeto==true && mover[]==true &&
arrastrar[]==true){
        Posición[][0]= ( Posicion[0][0]+ CursorX_final)-
CursorX_inicial;
        Posición[][1]=( Posicion[0][1]+ CursorY_final)-
CursorY_inicial;
        Bloque[].setPosicion( Posicion[][0], Posicion[][1]);
    }

    for(int n=1; n<numBloques; n++) {

```

```

        arrastrar_Objetos[n]=false;
        ...
    }

```

g) Método `mouseMoved`

Este método activa las variables `arrastrar[]` de todos los bloques permitiendo que se pueda arrastrar cualquiera.

```

mouseMoved(MouseEvent e) {
    for(int n=0; n<numBloques; n++){
        ...
        arrastrar_Objetos[n]=true;
        ...
    }

```

h) Método `mouseClicked`

El uso de este método permite que los bloques cambien de signo, de positivo a negativo y viceversa, y que cambien su orientación, de vertical a horizontal y viceversa.

```

mouseClicked ( MouseEvent e ){

```

Para girar los bloques se utiliza la siguiente sentencia:

```

    if(e.isShiftDown()&&estaDentro_ Objeto (e)){
        temporal=tamaño_x[];
        tamaño_x[]=tamaño_y[];
        tamaño_y[]=temporal;
        Bloque[].setSize(tamaño_x[],tamaño_y[]);
    }

```

El método *isShiftDown()* regresa verdadero si la tecla Shift es presionada, si se encuentra dentro de algún bloque, guarda e intercambia el tamaño de sus lados.

Para cambiar el signo de los bloques se utiliza la siguiente sentencia:

```
If (e.getClickCount()==2 && estaDentro_Objeto(e) ) {  
  
    if(Bloque[].getBackground ( )== colorPositivo ) {  
  
        Bloque[].setBackground ( colorNegativo );  
        nombre[].setText("-nombre");  
        Contador[]=-1;  
    }  
    else {  
        Bloque[].setBackground( colorPositivo );  
        nombre[].setText(" nombre");  
        Contador[]=1;  
    }  
    }  
    ...  
    }
```

El método *getClickCount()* devuelve el número de clicks del ratón, si es igual a dos y se encuentra dentro de algún bloque, revisa el signo del bloque y lo cambia, modificando también el nombre, color y contador asociado a este bloque.

i) Método *mouseReleased*

Este método verifica si los bloques son soltados dentro de cierta área de la aplicación, modificando la variable *soltar[]*.

```

mouseReleased ( MouseEvent e ) {
    ...
    if(estaDentro_Area(e) && estaDentro_Objeto (e) ) {
        soltar[]=true;
    }
    else

        if(!estaDentro_Area(e) && estaDentro_Objeto (e) ) {
            soltar[]=false;
        }

```

Después se asigna a la variable *contadorBloques* la suma del producto de las variables *contador[]* y el valor entero de *soltar[]*.

```

    ...
    contadorBloques = ( Contador[] ) * ( soltar[] ? 1 : 0 ) + ...)
    ...
}

```

j) Método LimiteBloque_XX

El método *LimiteBloque_XX* permite que el tamaño de los bloques *X* y *XY* puedan modificar su tamaño en *X*. El valor entero *a* es el tamaño mínimo del lado del bloque en *X*, y el valor del entero *b* es el tamaño máximo.

```

LimiteBloque_XX () {
    if (int a < tamaño_x[] < int b){
        return true;
    }
    else{
        ...
    }

```


k) Método LimiteBloque_YY ()

El método *LimiteBloque_YY* permite que el tamaño de los bloques *Y* y *XY* puedan modificar su tamaño en *Y*. El valor entero *a* es el tamaño mínimo del lado del bloque en *Y*, y el valor del entero *b* es el tamaño máximo.

```
LimiteBloque_YY ( ){  
    if (int a < tamaño_y[] < int b){  
        return true;  
    }  
    else{  
        ...  
    }  
}
```

CAPITULO V

CONCLUSIONES

La mayoría de los estudiantes de nivel básico y superior presentan dificultades en el área de álgebra, en particular, en la factorización de polinomios, es por ello que surgió la idea de desarrollar un software dinámico para ayudar en la enseñanza y el aprendizaje de este tema, dado que los entornos que existen actualmente en el mercado son de tipo matemático y no de tipo didáctico.

El desarrollo de este software no fue trabajo fácil ya que al inicio hubo dificultades en la interacción y visualización simultanea entre la representación algebraica y gráfica de los elementos. Una vez superada esta etapa, se optó por crear un software donde no sólo aparecieran algunos ejercicios básicos, sino que tanto el profesor como el estudiante tuvieran un rango amplio para la elección de los ejercicios a resolver.

El software se diseñó en etapas y en cada una se incorporaron secciones de lo particular a lo general, para que, al momento en que el alumno llegue al tema principal, la factorización de polinomios, esté familiarizado con los elementos del programa.

Los objetos matemáticos utilizados en el software se manipulan mediante el arrastre del puntero del ratón de una sección a otra, en una sección se muestra la representación algebraica del objeto y en otra se visualiza de manera gráfica. Éstos objetos se diseñaron para que puedan cambiar de tamaño mediante deslizadores y pueda ser representado el concepto de variable.

El software incluye dentro de una misma página las herramientas necesarias para escribir un polinomio, así como los bloques tanto negativos como positivos, para formar una figura rectangular que represente al polinomio, esto crea una gran ventaja ya que se tienen la parte gráfica y algebraica en un mismo sitio, lo que lleva al alumno a visualizar, descubrir y apropiarse de su conocimiento.

El software además de incorporar las herramientas antes mencionadas, también guía al estudiante en el manejo del software incorporando desde los primeros niveles guías en las actividades con los polinomios y es en la última sección donde no se incluyen, permitiendo que el alumno ponga en práctica el conocimiento adquirido en las secciones previas.

El potencial que ofrece el software además de abordar los temas de operaciones con polinomios es que se pueden trabajar temas como el conjunto de los números naturales, el conjunto de los números enteros, la resolución de ecuaciones de primer y segundo grado, así como algunas series numéricas de la teoría de números.

Cabe señalar que en desarrollo de esta aplicación no se realizaron todas las fases que integran la metodología descrita en el capítulo 2, pero proponemos más adelante llevar a cabo las fases restantes y llevar a las aulas esta aplicación la cual debe ir acompañada de actividades previamente planeadas por el profesor para maximizar el aprendizaje de los estudiantes.

BIBLIOGRAFÍA

- Adalid Diez de U, C. (1998). Fundamentos de algebra (1st ed., pp. 55-69). México, D.F.: Universidad Autónoma Metropolitana, Unidad Xochimilco, División de Ciencias Sociales y Humanidades.
- Arreaza, T. & Carvajal, E. (2013). Aritmética y álgebra a través de los bloques de Dienes (1st ed., pp. 1-10). Venezuela: Instituto Pedagógico de Caracas. Consultada de <http://cibem.semur.edu.uy/7/actas/pdfs/432.pdf>
- Clements, D. & Battista, M. (2000). Designing Effective Software. ResearchGate. Consultada el 13 agosto 2016, de https://www.researchgate.net/publication/258933241_Designing_effective_software
- Deitel, P. J. (2004). Cómo programar en Java. Pearson Educación.
- Dreyfus, T. (1994). The role of cognitive tools in mathematics education. En Biehler, R., et al. Didactics of Mathematics as a Scientific Discipline. Dordrecht: Kluwer, pp. 201-211.
- Duval, R. (2006). Un tema crucial en la educación matemática: La habilidad para cambiar el registro de representación. La Gaceta de la Real Sociedad Matemática Española, 9(1), 143-168.
- Kieran, C. (1981). Concepts associated with the equality symbol. Northern Arizona University. Consultada el 3 septiembre 2016, de <https://oak.ucc.nau.edu/smg224/401pdfs/algebrareadings/kieran1.pdf>
- Krishnamani, V. and Kimmons, D. (1994) Using technology as a tool in abstract algebra and calculus: The MTSU experience. Electronic proceedings of the 7th ICTCM, Orlando, FL. Consultada el 4 de noviembre de 2016 de: <http://archives.math.utk.edu/ICTCM/VOL07/C007/paper.pdf>
- Matemática Discreta usando Isetl. Fing.edu.uy. Consultado el 10 noviembre de 2016, de <https://www.fing.edu.uy/~darosa/curso.html>
- Medina, M. M. P. (1998). La adquisición del lenguaje algebraico y la detección de errores comunes cometidos en álgebra por alumnos de 12 a 14 años (Doctoral dissertation, Universidad de La Laguna).

- Saucedo, G. (2007). Categorización de errores algebraicos en alumnos ingresantes a la Universidad. *Itinerarios Educativos*, 1(2), 22-43.
- Schwarz, B., & Dreyfus, T. (1993). Measuring integration of information in multirepresentational software. *Interactive Learning Environments*, 3(3), 177-198.
- SEP 2015, Reforma Educativa, <https://www.gob.mx/sep/reformas/preguntas-frecuentes-de-la-reforma-educativa>, consultada el 8 de noviembre de 2016.
- Thomas, J., Ramos, O., & Maldonado, O. (2014). Patrones de Error que Presentan los Alumnos de Álgebra I en la Simplificación de Expresiones Algebraicas Racionales del I Período de la UPNFM, en la Modalidad Presencial de Tegucigalpa. *Revista Académica Paradigma Estudiantil*, 1(1), 35-45. Recuperado de <http://postgrado.upnfm.edu.hn/docs/vrip/repaes.pdf#page=36>