REVISTA ELECTRÓNICA DE INVESTIGACIÓN
EN EDUCACIÓN EN CIENCIAS
ISSN 1850-6666

Enseñanza de la programación en la Escuela Secundaria de Adultos a partir de un Recorrido de Estudio e Investigación

Angel Donvito, María Rita Otero

adonvito@exa.unicen.edu.ar , rotero@exa.unicen.edu.ar

Núcleo de Investigación en Educación en Ciencia y Tecnología (NIECyT)
Universidad Nacional del Centro de la Provincia de Buenos Aires (UNICEN)
Consejo Nacional de Investigaciones Científicas y Técnicas (CONICET)

Resumen

En este trabajo se propone un Recorrido de Estudio e Investigación (REI) para introducir la enseñanza de la programación en las clases de matemática de la Educación Secundaria de Adultos. La cuestión generatriz es ¿Cómo programar el robot Sphero para que forme con su trayectoria polígonos regulares y exprese sus principales características? Aquí se presenta un posible Modelo Praxeológico de Referencia referido al REI, donde se describen las sub-cuestiones derivadas y las praxeologías involucradas tanto de las ciencias de la computación como de matemática. El recorrido involucra nociones básicas de programación como comandos, variables, decisiones, bucles y a los polígonos regulares desde la geometría sintética y analítica. Los saberes computacionales y matemáticos involucrados poseen utilidad formativa inherente porque se requiere programar el robot, identificar las variables necesarias, realizar cálculos y establecer donde deben tomarse las decisiones que permitan elaborar los diferentes algoritmos posibles.

Palabras clave: Recorrido de Estudio e Investigación; Programación; Matemática; Modelo Praxeológico de Referencia.

Introduction to programming in adult secondary school through Study and Research Path

Abstract

In this work, a Study and Research Path (SRP) to introduce teaching of programming in mathematics classes in Adult Secondary Education is proposed. The SRP starts from the question how do to code a Sphero robot for it draws with its trajectory regular polygons and expresses its main features? We develop a Praxeological Reference Model and describe the derived sub-questions and the praxeologies involved in both computer science and mathematics. The study involves basic programming notions such as commands, variables, decisions, loops, and regular polygons from synthetic and analytic geometry. The results show that Computational and Mathematical knowledge could be studied for its inherent formative utility to program the robot: identify the necessary variables, perform calculations and establish the main decisions that may lead the algorithm by different procedures.

Keywords: Study and Research Path; Coding; Mathematics; Praxeological Reference Model.

Introduction à la programmation des écoles secondaires pour adultes à travers d'un Parcours d'Étude et de Recherche

Resumé

Dans ce travail, nous proposons d'étude et de recherche (PER) pour introduire l'enseignement de la programmation dans les classes de mathématiques de l'enseignement secondaire pour adultes. La question de parti du PER est: comment programmer un robot Sphero pour il dessine avec sa trajectoire des polygones réguliers et exprime ses principales caractéristiques? Nous développons un modèle praxéologique de référence et décrivons les sous-questions dérivées et les praxéologies impliquées à la fois en informatique et en mathématiques. L'étude implique des notions de programmation de base telles que les commandes, les variables, les décisions, les boucles, et des polygones réguliers issus de la géométrie synthétique et analytique,. Les résultats

Recepción:09/10/2020 *Aceptación: 10/11/2020*

montrent que les connaissances informatiques et mathématiques pourraient être étudiées pour leur utilité formative inhérente à la programmation du robot: identifier les variables nécessaires, effectuer des calculs et établir les principales décisions qui peuvent conduire l'algorithme par différentes procédures.

Keywords: Le parcours d'étude et de recherche; La programmation ; Les mathématiques; Modèle praxéologique de référence.

Introdução à programação na escola secundária para adultos por meio de um Percurso de Estudo e Pesquisa

Resumo

Neste artigo, propomos um Percurso de Estudo e Pesquisa (PEP) visando introduzir o ensino de programação em aulas de matemática no Ensino Médio de Adultos. O PEP começa com a pergunta: como programar o robô Sphero para que forme polígonos regulares com sua trajetória e expresse suas características principais? Desenvolvemos um modelo praxeológico de referência e descrevemos as subquestões derivadas e as praxeologias envolvidas na ciência da computação e na matemática. O estudo envolve noções básicas de programação, como comandos, variáveis, decisões e loops, além dos polígonos regulares nos contextos da geometria sintética e analítica. O conhecimento computacional e matemático é estudado pela sua utilidade formativa inerente à programação do robô, tais como: identificar as variáveis necessárias, realizar cálculos e estabelecer onde as decisões que conduzem ao algoritmo por diferentes procedimentos devem ser feitas.

Keywords: Percurso de Estudo e Pesquisa; Programação; Matemática; Modelo praxeológica de referência.

1. INTRODUCCIÓN

La programación y la robótica son dos de los campos que experimentaron mayor auge en la última década. Se las considera de vital importancia para el desarrollo económico y social de una nación ya que permea casi todos los campos laborales. Así, desde el ministerio de educación argentino (CFE Nº 263/15) se establecieron los primeros lineamientos para desarrollar una enseñanza orientada a enseñar estos saberes. Sin embargo, aunque se han enunciado los Núcleos de Aprendizajes Prioritarios (NAPs) para la educación primaria y secundaria (CFE, 343/18), estos no se han adaptado a la educación de adultos. La Educación Secundaria de Adultos (ESA), presenta diferencias significativas con la educación secundaria obligatoria (ESO), tales como una disminución considerable del tiempo didáctico y de saberes a estudiar y diferentes intereses de los estudiantes sobre su formación. En estudios anteriores identificamos que los estudiantes de la ESA reclaman estudiar saberes que resulten de utilidad para la vida adulta y particularmente para el mundo del trabajo (Donvito, Otero, 2019a; 2019b). Dada esta demanda y ante la falta de un espacio curricular para enseñar programación en la ESA y a la carencia de equipamiento informático en estas instituciones, en este trabajo se propone un posible dispositivo didáctico para introducir a la programación en las clases de matemática. Se trata de un Recorrido de Estudio e Investigación (REI) codisciplinar, en ciencias de la computación y matemática, enmarcado en la Teoría Antropológica de lo Didáctico (TAD).

La TAD establece que la forma tradicional de enseñar matemáticas y otras ciencias, responde al denominado *paradigma monumental* o *de visitar a las obras* (Chevallard, 2013a, 2017). La metáfora del monumento, alude a que el saber se presenta como una obra expuesta en un museo, que se debe visitar, admirar, disfrutar y venerar, porque alguna vez tuvo (aunque ya no) alguna utilidad. El monumentalismo predominante en la cultura escolar actual, produce la *pérdida de sentido* y de las *razones de ser* del saber. Según la TAD, la razón de ser de los saberes se podría recuperar si los programas de estudios estuvieran compuestos de pares de preguntas Q_i y respuestas R_i (Chevallard, 2007). Se trataría de estudiar preguntas cuya respuesta además de no ser lineal ni trivial, requiera aprender lo necesario de ciertas organizaciones, por ejemplo, matemáticas, del currículo sin pretensión de agotar el estudio. Desarrollar una enseñanza de estas características supone cambios muy grandes con relación a las prácticas actuales (Otero, Fanaro, Corica, Llanos, Sureda, Parra, 2013; Salgado & Otero, 2020; Parra, Otero, 2018). Cambios, que implican una revolución epistemológica y didáctica, que motorice la sustitución del paradigma dominante por otro, llamado paradigma de la *Investigación y Cuestionamiento del Mundo (PICM)* (Chevallard, 2013a). Los Recorridos de Estudio e Investigación (REI), son dispositivos didácticos particularmente apropiados para enseñar en el paradigma del cuestionamiento, que aún están en etapa experimental.

Un REI es un artefacto didáctico o un instrumento, diseñado a partir de una pregunta generatriz, esto es, de una pregunta que origina y motiva tanto el estudio como la investigación y convoca a distintas praxeologías, que pueden ser matemáticas o extra matemáticas. La principal responsabilidad en el desarrollo de tales herramientas recae en los didactas y es fruto de la investigación didáctica como esta.

La TAD reformula la concepción de la didáctica y de lo didáctico al asumir un principio antropológico y epistemológico. La condición epistemológica implica que el análisis didáctico es indisociable del saber. La condición antropológica asume que un saber es siempre un producto de la cultura humana, establecido a partir de prácticas institucionales que lo caracterizan y lo legitiman, en relación con una actividad realizada en una cierta institución social en donde ese saber "vive" y se desarrolla (Otero, 2020). En consecuencia, la organización de la enseñanza de un cierto saber, se realiza siempre recurriendo a ciertos dispositivos más o menos explícitos. En el caso de los REI, su diseño requiere de un ejercicio dialéctico que supone analizar el conjunto de saberes que estarán involucrados en la elaboración de la respuesta a la pregunta generatriz, ampliándolo y escudriñándolo, así como la consideración de qué parte será profundizada, ya que es la pregunta la que motiva el estudio. Las preguntas de naturaleza epistémica son inseparables de las didácticas: ¿cómo estudiar una dada pregunta Q? Esto recuerda a Richard Feymmann, quien afirmaba que, para saber de algo, es preciso tener que enseñarlo.

La condición antropológica también establece que no existe un punto de vista o un sistema de referencia privilegiado desde donde observar los hechos didácticos. El saber matemático u otro, que vive en la escuela, no puede ser legitimado, valorado, ni entendido solo con referencia al saber sabio de los matemáticos profesionales. Debido a la relatividad institucional de los saberes, para desarrollar un REI o cualquier otro dispositivo didáctico, el didacta debe construir un sistema de referencia explícito llamado modelo praxeológico de referencia (MPR), que, por el principio de relatividad, es siempre provisorio y cuestionable

En este trabajo, presentamos un posible MPR de un REI que se desarrolla alrededor de la pregunta Q_0*: ¿Cómo programar el robot Sphero para que forme con su trayectoria polígonos regulares y exprese sus principales características?*

2. ESTADO DEL ARTE SOBRE LOS REI

Existen algunos trabajos donde se describe el proceso de diseño, de implementación y/o de análisis de distintos REI para la educación secundaria y la universidad. En un trabajo realizado por Parra y Otero (2018) se releva la existencia de 41 publicaciones científicas al respecto y se describen las principales características de cada una. Se registraron REI mono-disciplinares en matemática o multidisciplinares, relacionados con trece disciplinas como por ejemplo economía, física, historia, medicina nuclear, entre otras.

El MPR que presentamos aquí está diseñado para la ESA y en este sentido es original, tanto por la institución como porque involucra a las Ciencias de la Computación y la Matemática.

Considerando a las matemáticas, este REI involucra el estudio de nociones geométricas básicas. En esta área encontramos solo dos trabajos referidos a REI mono disciplinares: la investigación de Gaud & Minet (2009) quienes dada la pregunta *¿Cómo construir una figura geométrica que se ajuste a ciertas condiciones?* proponen posibles cuestiones derivadas junto con las tareas y técnicas para su resolución; y la investigación de Quijano & Corica (2017) que desarrollan un MPR sobre los lugares geométricos. Resulta evidente que la geometría ha ido perdiendo terreno en la escuela. Por lo general, esto se le atribuye, entre otras razones, a la contracción del tiempo didáctico en conjunción con una priorización de otros saberes; a ciertas deficiencias en la formación docente en esta área; a la dificultad para encontrar problemas que presenten verdaderos desafíos; y a la escasa claridad de las nociones geométricas en los diseños curriculares. En el caso particular de la ESA, los estudiantes consideran poco útil a la geometría. Una encuesta sobre 820 casos muestra que los estudiantes adultos asignan escasa utilidad para la vida a la geometría de la ESA (2%), así como para el trabajo (2%) y para estudiar una carrera (4%) (Donvito, 2018; Donvito, Otero, 2019a, 2019b).

3. LOS REI

Al comenzar un REI el profesor introduce una pregunta Q en sentido fuerte, llamada generatriz, cuyo estudio conducirá al encuentro o reencuentro con praxeologías de una o de varias disciplinas, en este caso: la Matemática y las Ciencias de la Computación. El estudio de Q genera una cadena de preguntas y respuestas, que son el corazón del proceso de estudio (Chevallard, 2007). La "vida" de la pregunta generatriz Q y la elaboración de una posible respuesta, requieren construir un medioambiente apropiado y una infraestructura para desarrollar y validar una respuesta posible – un medio didáctico M. Dicho ambiente es fabricado y organizado por la comunidad de estudio. El medio no está construido previamente, sino que se va generando a medida que se va elaborando una respuesta a Q, a partir de todos los elementos que se decide incorporar a él, en la medida en que serían potencialmente útiles para elaborar las respuestas. Construir M es una responsabilidad de los estudiantes y del profesor, siendo este último solo un sistema de información más de la clase, es decir que sus informaciones reciben idéntico tratamiento que las de otros, no rige un principio de autoridad.

3.1. Praxeología

La noción de *praxeología* (Chevallard, 1999) es una expresión cabal de la condición antropológica de la TAD. La palabra compuesta *praxeología* designa a un modelo único que permite modelar cualquier actividad humana regularmente organizada. Una praxeología contiene dos bloques: uno relativo a la práctica o praxis $[T/\tau]$, y otro referido a un cierto nivel más o menos desarrollado de discurso sobre la praxis, al que se denomina logos $[\Theta / \theta]$. Este modelo de las praxeologías es general y se refiere a cualquier ámbito o dominio de la actividad humana, ya que se requiere de una praxeología para resolver una ecuación, lavarse las manos, componer una canción, hacer las compras en el supermercado o preparar el desayuno. Este tipo de actividades requieren de técnicas que se usan para realizar tal o cual tipo de tarea, y algún nivel de discurso justificativo de ellas, por mínimo que este sea. Una praxeología es una 4-upla, que se escribe $[T/\tau \ / \ \Theta \ /\theta]$ integrada por cuatro componentes: un tipo de tareas T; una técnica τ, es decir una manera de realizar las tareas del tipo T; una tecnología θ, es un discurso racional sobre la técnica τ que pretende justificarlo, hacerlo legítimo e inteligible; y una teoría Θ, que permite generar, legitimar y justificar θ.

Es destacable que también suele llamarse a las praxeologías con la palabra que se usa en francés para referirse a un trabajo u ocupación *oeuvre*, obra en español. Entonces una obra o praxeología es entendida como una creación humana cuyo objetivo original es, o ha sido, ser útil para responder cierta pregunta. En este contexto, las obras no se valoran como "nobles", "loables, "o recomendables", sino por su funcionalidad o utilidad en algunas situaciones e instituciones.

3.2. La utilidad formativa del saber

Desde una perspectiva axiológica, la TAD establece que el saber a enseñar tiene una utilidad formativa inherente, siendo este un valor epistémico en el sentido de Laudan (1984). La perspectiva antropológica adopta una postura funcionalista e instrumental de las matemáticas a las que considera como una más, entre otras actividades humanas. Esta actividad produce *obras* de la cultura o praxeologías matemáticas, que surgen frente a la necesidad de resolver y responder cuestiones vitales. Existen dos tipos de *utilidad formativa* del saber a enseñar: *trascendente* e *inherente* (Chevallard, 2017; Kim, 2015; Donvito, 2018). La primera va más allá de aquello para lo que sirve intrínsecamente un saber, por ejemplo, suele justificarse el estudio de las propiedades de los polígonos regulares y los teoremas asociados porque permiten “desarrollar el pensamiento lógico-deductivo y el rigor matemático”. La *utilidad formativa inherente* en cambio, consiste en “el dominio del saber de manera que pueda utilizarse en situaciones de la vida social” (Kim, 2015, p. 274). De esta manera, la programación, los polígonos regulares y sus propiedades son importantes porque permiten enfrentar problemas relativos al mundo de la vida (Habermas, 1987a, 1987b).

4. EL AUTÓMATA UTILIZADO EN ESTE REI

En esta investigación se optó por el robot *Sphero Bolt* (Figura 1) que es un dispositivo programable envuelto en una esfera de policarbonato, capaz de rodar en distintas direcciones con una precisión suficiente para trazar con su trayectoria figuras geométricas. Debido a esta característica, lo consideramos como una herramienta útil para realizar proyectos de enseñanza multidisciplinar, especialmente relacionados a la matemática, la física y la programación. Este autómata es adecuado para nuestro objetivo por dos razones: la primera es que se programa mediante un software compatible con cualquier Smartphone que cuente con bluetooth. Esto permite prescindir de aulas especiales, de “laboratorios de informática” e incluso de computadoras. Los estudiantes pueden utilizar sus teléfonos móviles para programar, guardar, repasar y reescribir los programas dónde y cuándo quieran. La segunda razón es que la interfaz del dispositivo soporta tanto un *lenguaje de programación por bloques* como un *lenguaje de programación codificada*. En el contexto de la ESA, la programación por bloques resulta adecuada para introducir las estructuras y los comandos básicos de programación en un tiempo mucho más breve que sí utilizáramos programación codificada.

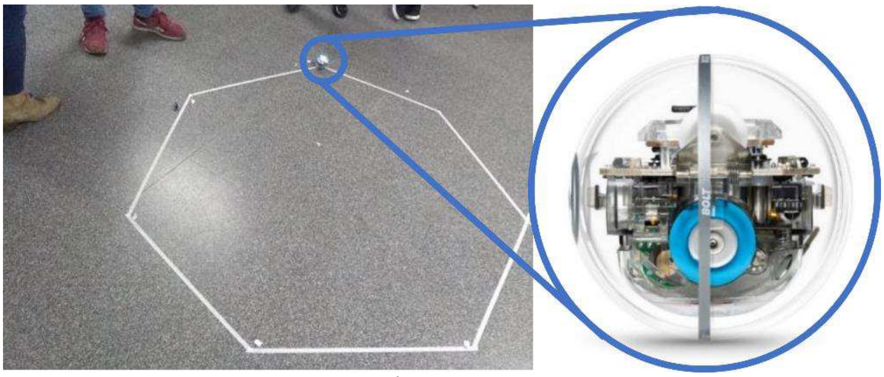


Figura 1. El autómata: Sphero Bolt[1] e imágenes del trabajo de campo en la ESA

5. EL MODELO PRAXEOLÓGICO DE REFERENCIA

Las praxeologías nacen, viven, mueren y se desarrollan en una institución social que las produce y las legitima, en este caso la ESA. La relatividad institucional del saber es un principio fundacional y definidamente antropológico de la TAD. Este principio implica la imposibilidad de adoptar una referencia absoluta para analizar los hechos didácticos, tanto en la posición de investigador como de enseñante. De allí la necesidad de construir un modelo de referencia que permita identificar las praxeologías involucradas en el proceso de enseñanza y de estudio, así como los caminos que conducen a ellas. En este sentido, un *Modelo Praxeológico de Referencia (MPR)* (Chevallard, 2013b) de un REI, como el que se presenta a continuación, consiste en un análisis profundo de las preguntas y las praxeologías de las distintas disciplinas involucradas, desde la perspectiva del didacta.

A modo de síntesis, en la Figura 2 se presenta el conjunto de sub preguntas y praxeologías derivadas de la pregunta Q_0. Las praxeologías involucran nociones de al menos dos disciplinas. Por un lado, organizaciones matemáticas (OM) referidas a las propiedades y construcción de polígonos regulares y por otro, praxeologías básicas del ámbito de las Ciencias de la Computación (OCC) para programar el autómata.

El estudio de Q_0 genera al menos cuatro preguntas derivadas. Por un lado, dentro de las Ciencias de la Computación una de las cuestiones derivadas es: Q_1*: ¿Cómo establecer algoritmos para realizar tareas?* Aquí surge la OCC1que se define a partir del tipo de tarea T_1*: Establecer algoritmos para realizar tareas específicas.* Esta pregunta nos introduce a las praxeologías computacionales. Por su parte, como el problema involucra un robot específico, surge la pregunta Q_2*: ¿Cómo programar un robot Sphero?* Así, el estudio involucra a la OCC2, que se define a partir del tipo de tareas T_2*: Programar el robot Sphero*.

Por otro lado, en el ámbito de las matemáticas, surge Q_3*: ¿Cuáles son las características de los polígonos regulares?* Esta pregunta conduce a la OM1 y a la OM2. La primera se encuentra definida por el Tipo de Tarea T_3*: Establecer propiedades métricas de los polígonos regulares*. Esto da lugar al estudio de los polígonos regulares en la geometría sintética. La OM2 está definida por el Tipo de Tarea T_4*: Establecer propiedades analíticas de los polígonos regulares*. Esta última, conduce al estudio de los polígonos regulares en la geometría analítica.

El estudio matemático se completa con la pregunta derivada Q_4*: ¿Cómo construir un polígono reglar?* que conduce a las organizaciones matemáticas OM3 y OM4. Aquí, se propone integrar las nociones y propiedades métricas y analíticas que

[1] https://sphero.com/pages/educators

emergen del estudio de Q_3. El tipo de tareas que definen a la OM3 es T_5: *Representar polígonos regulares en el plano.* Para esto, es preciso el uso de instrumentos de geometría, cuyas tecnologías se basan en propiedades métricas. Por otro lado, el tipo de tareas que integra a la OM4 es T_6: *Representar polígonos regulares en el plano cartesiano* requiere aplicar técnicas analíticas cuya tecnología podría recuperar algunos fundamentos métricos, pero se caracteriza principalmente por el empleo de técnicas algebraicas.

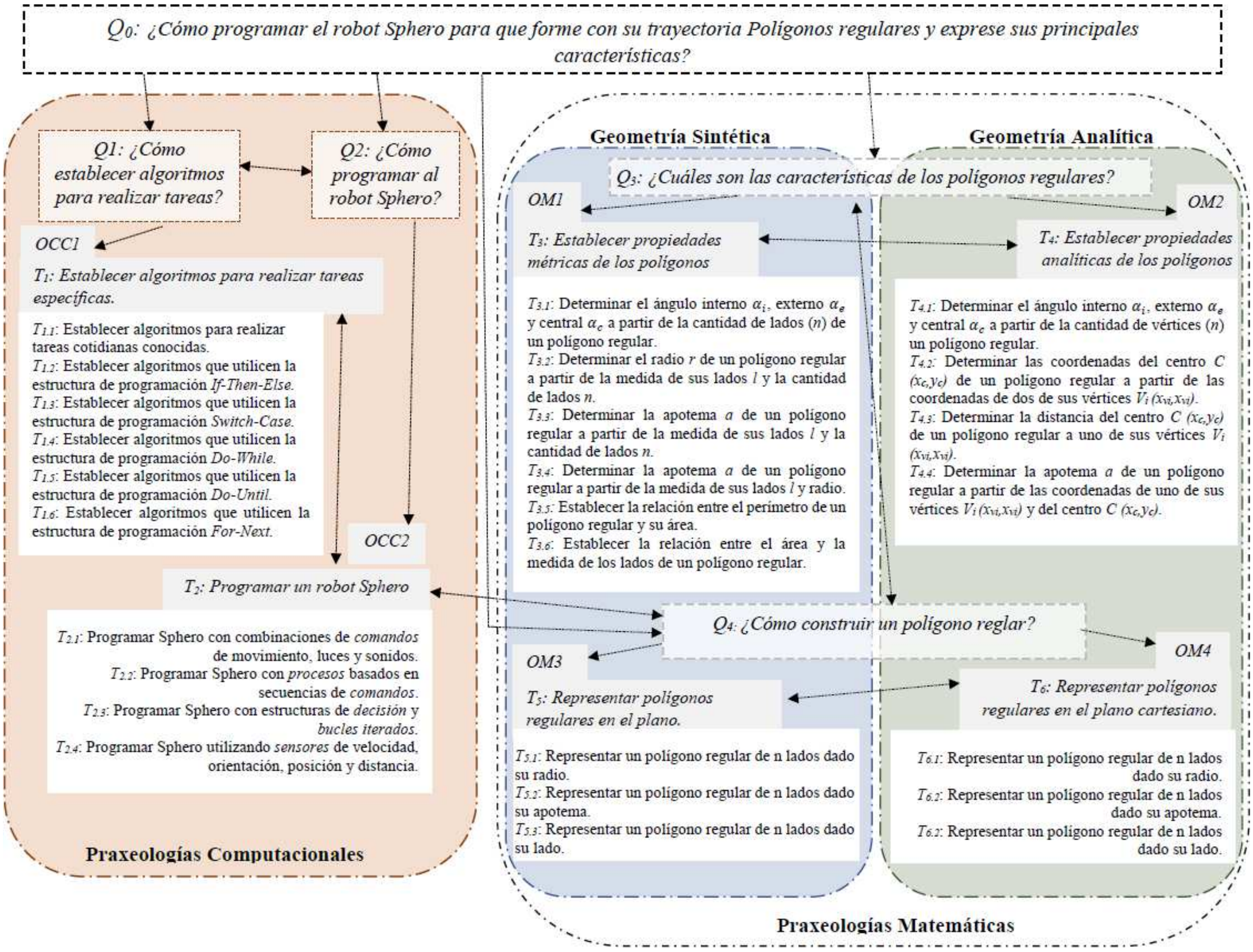


Figura 2: Esquema del MPR

Organización de Ciencias de la Computación 1 (OCC1)

La OCC1 se define a partir del tipo de tarea T_1: *Establecer algoritmos para realizar tareas específicas.* De manera general los *algoritmos* son los procedimientos metódicos que subyacen en la adquisición, representación, procesamiento, almacenamiento, comunicación y acceso a la información. Existen algoritmos para casi todo, para preparar un té, para llegar de la casa al trabajo, para realizar una división entre dos números naturales, para resolver una ecuación, etc. Particularmente nos interesan aquí los algoritmos matemáticos vinculados con el problema y con algún tipo de esquematización y comunicación de los procesos involucrados, para sentar las bases que permitan adentrase luego en la programación del autómata. En este último aspecto, se deben conocer las características de las órdenes o comandos, tales como precisión y secuenciación clara. También es preciso conocer los distintos tipos de variables más comunes y los procesos a los que se las puede someter: las textuales, que se pueden leer; numéricas (reales o enteras) con las que se puede operar; y las booleanas que comprenden los valores verdadero y falso. Luego de conocer esto, las tareas a realizar aquí serían:

$T_{1.1}$: Establecer algoritmos para realizar tareas cotidianas conocidas.

$T_{1.2}$: Establecer algoritmos que utilicen la estructura de programación *If-Then-Else.*

$T_{1.3}$: Establecer algoritmos que utilicen la estructura de programación *Switch-Case.*

$T_{1.4}$: Establecer algoritmos que utilicen la estructura de programación *Do-While.*

$T_{1.5}$: Establecer algoritmos que utilicen la estructura de programación *Do-Until.*

$T_{1.6}$: Establecer algoritmos que utilicen la estructura de programación *For-Next.*

Organización de Ciencias de la Computación 2 (OCC2)

Una vez estudiadas las nociones básicas de los algoritmos, lo siguiente es interiorizarse en la programación del robot *Sphero.* Aquí es importante reconocer la diferencia entre algoritmo y programa. Si bien el programa coincide con el algoritmo en intentar establecer un conjunto de procedimientos, lo hace a partir de un lenguaje específico, para un autómata particular y disponiendo solamente de los comandos que éste puede procesar. Por lo tanto, hay que explorar las posibilidades de programación de nuestro autómata particular. El tipo de tareas aquí es T_2: *Programar un robot Sphero*:

$T_{2.1}$: Programar Sphero con combinaciones de *comandos* de movimiento, luces y sonidos.

$T_{2.2}$: Programar Sphero con *procesos* basados en secuencias de *comandos*.
$T_{2.3}$: Programar Sphero con estructuras de *decisión* y *bucles iterados.*
$T_{2.4}$: Programar Sphero utilizando *sensores* de velocidad, orientación, posición y distancia.

Organización Matemática 1 (OM1)

La OM1, definida por el Tipo de Tarea T^3*: Establecer propiedades métricas de los polígonos regulares*, se desarrolla en el marco de la geometría sintética. Aquí, luego de identificar y definir los elementos de los polígonos regulares, se realizan distintas sub-tareas relacionadas con determinar cada uno de esos elementos en función de los otros:

$T_{3.1}$: Determinar el ángulo interno α_i, externo α_e y central α_c a partir de la cantidad de lados (n) de un polígono regular.
$T_{3.2}$: Determinar el radio *r* de un polígono regular a partir de la medida de sus lados *l* y la cantidad de lados *n.*
$T_{3.3}$: Determinar la apotema *a* de un polígono regular a partir de la medida de sus lados *l* y la cantidad de lados *n.*
$T_{3.4}$: Determinar la apotema *a* de un polígono regular a partir de la medida de sus lados *l* y radio.
$T_{3.5}$*:* Establecer la relación entre el perímetro de un polígono regular y su área.
$T_{3.6}$: Establecer la relación entre el área y la medida de los lados de un polígono regular.

Organización Matemática 2 (OM2)

La OM2, definida por el Tipo de Tarea T^4*: Establecer propiedades analíticas de los polígonos regulares,* cobra sentido y utilidad para estudiar Q_0 dado que el autómata devuelve como output el recorrido en un plano cartesiano. Los sensores del robot permiten crear un sistema referencia de la posición en función del tiempo. En el plano cartesiano se identifican las coordenadas de cada uno de los vértices, lo que permite realizar los mismos tipos de tareas que en la OM1, pero a partir de las coordenadas de los vértices del polígono.

$T_{4.1}$: Determinar el ángulo interno α_i, externo α_e y central α_c a partir de la cantidad de vértices (n) un polígono regular.
$T_{4.2}$*:* Determinar las coordenadas del centro $C\ (x_c,y_c)$ de un polígono regular a partir de las coordenadas de dos de sus vértices $V_i\ (x_{vi},x_{vi})$.
$T_{4.3}$: Determinar la distancia del centro $C\ (x_c,y_c)$ de un polígono regular a uno de sus vértices $V_i\ (x_{vi},x_{vi})$.
$T_{4.4}$: Determinar la apotema *a* de un polígono regular a partir de las coordenadas de uno de sus vértices $V_i\ (x_{vi},x_{vi})$ y del centro $C\ (x_c,y_c)$.

Organización Matemática 3 (OM3)

La OM3 se define a partir del tipo de tarea T^5*: Representar polígonos regulares en el plano.* Dentro del marco de la geometría sintética se estudia cómo realizar la representación gráfica de los polígonos regulares según sus características. Aquí se propone el trazado de los polígonos regulares utilizando elementos como regla, compás y transportador, para lo cual es necesario recuperar las nociones de la OM1.

$T_{5.1}$: Representar un polígono regular de n lados dado su radio.
$T_{5.2}$: Representar un polígono regular de n lados dado su apotema.
$T_{5.3}$: Representar un polígono regular de n lados dado su lado.

Organización Matemática 4 (OM4)

La OM4 se define a partir del tipo de tarea T^6*: Representar polígonos regulares en el plano cartesiano.* Dentro del marco de la geometría analítica se estudia cómo realizar la representación gráfica de los polígonos regulares según sus características. Aquí se propone el trazado de los polígonos regulares con a partir de pares *(x,y)*, para lo cual es necesario recuperar las nociones de la OM2.

$T_{6.1}$: Representar un polígono regular de n lados dado su radio.
$T_{6.2}$: Representar un polígono regular de n lados dado su apotema.
$T_{6.2}$: Representar un polígono regular de n lados dado su lado.

5.1. Descripción del MPR

El REI parte de la pregunta Q_0*:¿Cómo programar el robot Sphero bolt para que forme con su trayectoria Polígonos regulares y exprese sus principales características?* Las cuestiones vinculadas a un autómata, el tipo de órdenes que puede procesar y su lenguaje de programación, por más rudimentario que este fuera, son las primeras preguntas a abordar. Hay que estudiar cómo se programan y reconocen las variables involucradas en la programación, antes del modelado matemático del problema. De esta manera, las primeras tareas apuntarían a estudiar pragmáticamente los comandos básicos del autómata, las estructuras de programación por pilas de bloques y el tipo de órdenes que procesan: ordenadas y precisas. Cada uno de los elementos de la *app* se introducen y se utilizan en la programación de maneras diferentes según su género: *comandos* de *luces*, *sonidos* o *movimientos*, *estructuras* para crear *procesos*, comandos para operar *variables* y para establecer *bucles* iterados, entre otros.

Los *comandos de movimientos* se introducen asignando un sentido, una velocidad y el tiempo de su ejecución. Las variables involucradas de tipo *float* son: la orientación *O* en sentido horario*,* con $0° < O < 360°$*;* la velocidad *v,* con $-255 < v < 225$; y la duración (o tiempo) *t* en segundos*,* con $0 < d < 999999$. La Figura 3, muestra como cargar en la interfaz un movimiento con *O=90*, *v=100* y *t=3.*

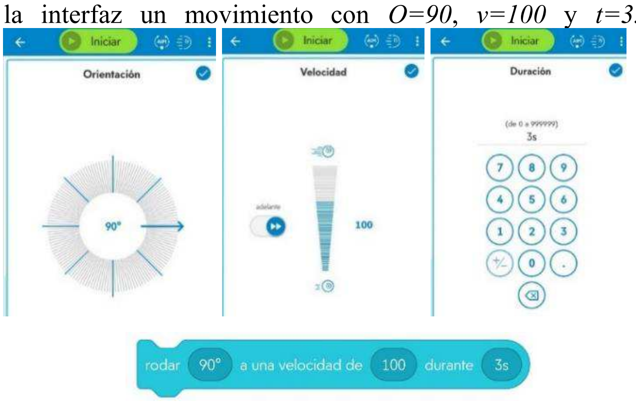


Figura 3. Comandos de movimiento

El programa soporta dos tipos de *comandos de sonido*, por un lado, sonidos precargados de distintos tipos y por otro, la lectura de un texto escrito por el usuario. En ambos casos, se selecciona si el programa debe "esperar" hasta que termine de reproducirse el sonido antes de pasar al siguiente bloque o si debe "continuar" con el siguiente comando inmediatamente. La Figura 4 muestra dos bloques, el primero reproduce un sonido de moneda mientras que el segundo lee el consabido texto *"hola mundo"*.

Figura 4. Comandos de sonido

Los *comandos de luces* permiten encender, apagar, parpadear, atenuar y cambiar de color las luces led del autómata. La versión que utilizamos tiene además una matriz de (8x8) leds, que permite representar imágenes simples, mostrar animaciones e incluso un texto desplazable. La Figura 5 presenta un bloque para mostrar en la matriz led el texto desplazable "hola mundo", a veinte cuadros por segundo.

Figura 5. Comando de texto desplazable

La programación del autómata soporta también diferentes bucles relacionados a las estructuras de control (Figura 6): repetir n veces el bucle (*For-Next*); repetir hasta que se cumpla una condición (*Do-Until*); si es verdadera una condición entonces entrar al bucle (*Do-While*); llamar a un conjunto de acciones si una condición es verdadera y de caso contrario a otro (*If-Then-Else*).

Figura 6. Estructuras de control mediante bucles

Como en cualquier lenguaje de programación, se pueden utilizar *operadores* de*:* suma, resta, multiplicación, división, operaciones trigonométricas, etc.; y *comparadores*: igual, distinto, mayor o igual, etc.; también se pueden crear *procesos* o funciones basadas en una serie de comandos. El autómata cuenta además con *sensores*, que arrojan valores numéricos *float* para indicar, entre otros, la velocidad, el espacio, el tiempo y el sentido. En conjunto, estos elementos permiten programar al autómata para que, en este caso, realice movimientos con trayectorias poligonales, y registre, analice y muestre las principales características de la figura trazada.

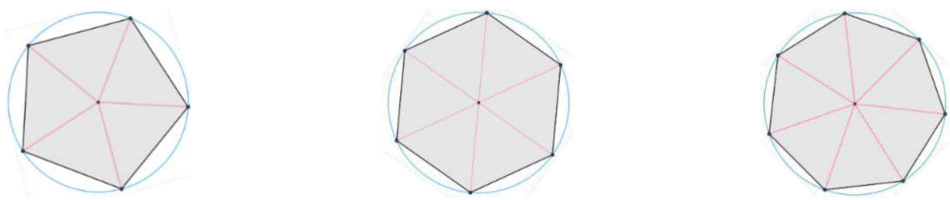
El dominio de las técnicas relativas a cada uno de los elementos mencionados del autómata requiere de tiempo y de varias situaciones. Por otro lado, la precisión del autómata, así como conocer sus comandos propios, es condición necesaria pero no suficiente para trazar polígonos regulares con sus trayectorias, los movimientos tienen que cumplir dos condiciones:

1- Cada movimiento debe ser de igual longitud

2- Luego de n movimientos similares, el autómata debe llegar al lugar de donde partió formando una figura cerrada.

Para alcanzar la condición 1, basta con fijar una misma velocidad y tiempo para cada movimiento. Sin embargo, la condición 2 requiere establecer los ángulos con precisión para que el movimiento se produzca en el sentido correcto. Aquí es cuando entra en juego la matemática y se aprecia la relevancia de estudiar los *polígonos regulares*, no de manera tradicional y con la profundidad necesaria y suficiente para establecer cuántos grados debe girar el robot al llegar a un vértice, según el polígono del cual se trate. Es decir, antes de poder establecer un algoritmo informático es necesario generar un algoritmo matemático. En esta instancia, la opción más apropiada es adoptar el marco de la *geometría sintética*, para estudiar las propiedades de los polígonos regulares y la relación entre sus ángulos.

Los polígonos regulares tales como el pentágono, el hexágono, el heptágono, etc. se encuentran, por definición, inscriptos en una circunferencia. Es decir, que todos sus vértices se intersecan con una circunferencia como se observa en la Figura 7.



Pentágono Hexágono Heptágono

Figura 7. Algunos polígonos regulares

Los ángulos internos de los polígonos regulares α son iguales y se calculan como α *=360/n,* siendo n el número de lados. A su vez, ya que está formado por dos radios y un lado, cada triangulo interior de un polígono regular es isósceles (o equilátero en el caso del hexágono). Entonces, la relación entre el ángulo central α y los ángulos del triángulo β y γ esta dada por $\beta = \gamma = \frac{180-\alpha}{2}$ (Figura 8). Además, el ángulo interior de un polígono regular es $\delta = 2\beta$, entonces $\delta = 180 - \alpha$.

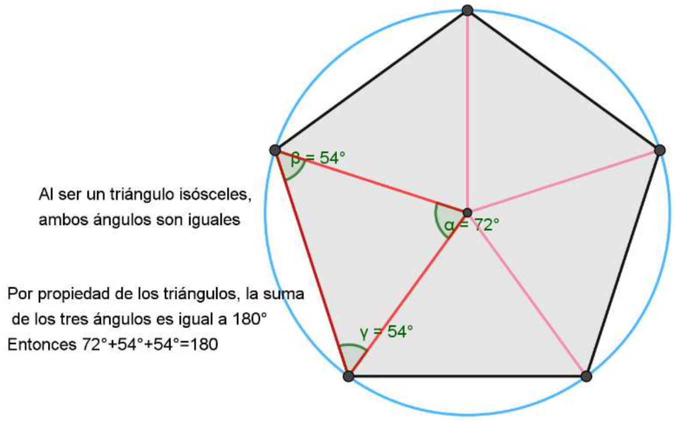


Figura 8. Procedimiento para calcular los ángulos centrales, internos y externos de un polígono

Una vez conocidas las relaciones entre los ángulos de los polígonos regulares, interesa saber ¿Cuánto debe doblar el autómata en cada vértice? Si se traza una semirrecta como prolongación de un movimiento del robot como muestra la Figura 9, el ángulo que debe girar Sphero es el suplemento del ángulo interno del polígono regular $180° - \beta - \gamma$, lo que por definición es el ángulo central α.

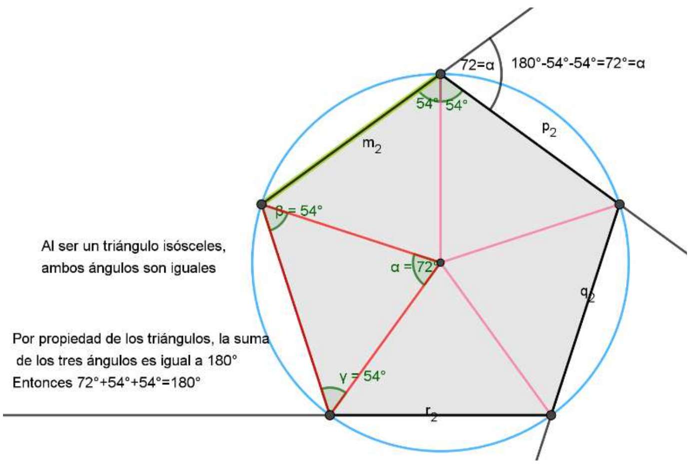


Figura 9. Procedimiento para hallar el ángulo de giro del autómata

Finalmente, como α depende de *n,* para saber cuánto debe girar Sphero para que realice trayectorias con forma de polígonos regulares, basta dividir $360/n$. Así, los únicos datos a ingresar en el programa son la cantidad de lados *n,* la velocidad del movimiento *v, y* la duración *t.* La Figura 10 muestra cómo adaptar este algoritmo a la programación por bloques. Aquí se utiliza un bucle de repetición *For-Next* que depende del valor *n* ingresado. El tamaño de los lados del polígono estará definido por *v* y *t* en cada comando "rodar". Es importante destacar que el retraso de 2 segundos luego de cada movimiento, permite al autómata detenerse y marcar bien los vértices, mientras recupera estabilidad antes de pasar al siguiente movimiento. No existe una única solución posible.

Figura 10. Programa para que Sphero se mueva trazando un polígono regular

Sería posible agregar a la programación, procesos que además de formar polígonos regulares con su trayectoria, arrojen como un output del autómata datos tales como la cantidad de lados, el valor de los ángulos interno y externo, la longitud de sus lados, la medida del perímetro, del radio, de la apotema, y el área. Los valores de sus ángulos internos y externos e incluso la suma de los ángulos de un polígono regular pueden ser calculados en función de *n* como se mostró anteriormente. Por lo que la primera parte del programa en Sphero podría desarrollarse como se muestra en la Figura 11.

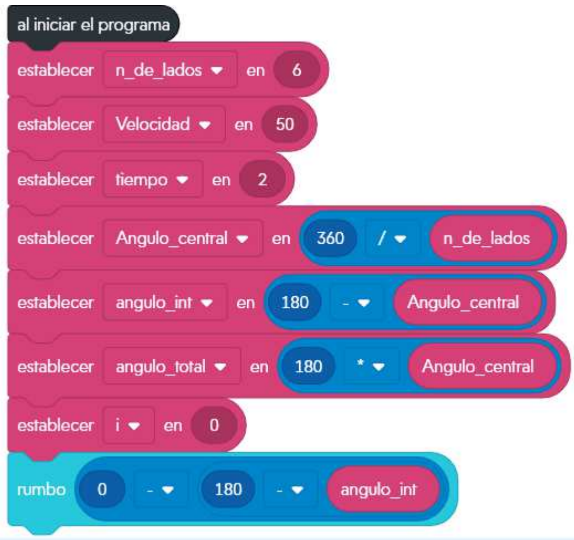


Figura 11. Programa en Sphero – primera parte

Las tareas relacionadas con hallar el valor del radio, la apotema y el área pueden ser realizadas en el marco de *la geometría sintética* como se vino haciendo hasta aquí, o en el de la *geometría analítica*. Para este último, el autómata registra el recorrido realizado y lo representa en un eje cartesiano, con suficiente precisión como para hacer los cálculos. Analíticamente la apotema puede hallarse de dos maneras, según *n* sea par o impar.

Si *n* es par, la posición del autómata antes de ejecutar el movimiento $\frac{n}{2} + 1$, es la posición del vértice E en el plano (Figura 12). Considerando el punto $E\ (x_E, y_E)$ se tiene que $2\,.\,a = x_E$. Luego, el radio se halla por medio del teorema de Pitágoras $r^2 = a^2 + (l/2)^2$, siendo *a* la apotema*,* y *l* el lado. Entonces $r = \sqrt{a^2 + \frac{l^2}{2}}$.

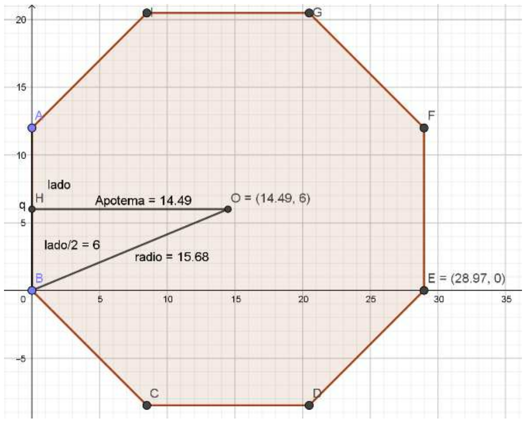


Figura 12. procedimiento para polígonos de n lados, con n par

Si *n* es *impar*, se guarda como una variable a la posición del autómata previa al movimiento $\frac{n-1}{2}+1$, que en la Figura 13 se muestra como el punto *E*. Entonces el segmento formado por el punto medio de $\overline{BA}$ y el punto *E* es igual a la suma de la apotema más el radio. El valor de la componente x_E de la posición $E\ (x_E, y_E)$ es el resultado de dicha suma $x_E = a + r$. Además, se puede hallar *r* en función de la apotema y el lado ya que $r^2 = a^2 + (l/2)^2$. Si se resuelve el sistema mixto de dos ecuaciones con dos incógnitas:

$$\begin{cases} x_E = a + r \\ r^2 = a^2 + (l/2)^2 \end{cases}$$

Se obtiene $r = \frac{x_E{}^2 + (l/2)^2}{2.x_E}$ y $a = x_E - \frac{x_E{}^2 + (l/2)^2}{2.x_E}$.

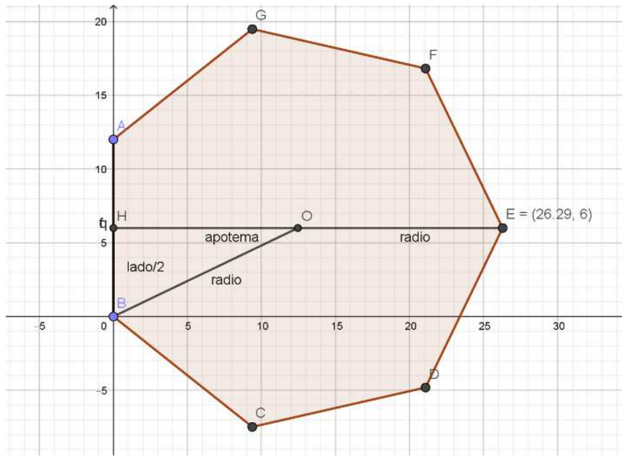


Figura 13. procedimiento para polígonos de n lados, con n impar

Este trabajo matemático revela la importancia de identificar si la cantidad de lados del polígono es par o impar para decidir el procedimiento para hallar la apotema y el radio. Entonces, el programa iniciado en la Figura 11, se continúa en la Figura 14. Se utiliza un bucle *For-Next* que depende del valor *n* ingresado y un índice *i,* para reconocer sobre qué vértice se posiciona Sphero luego de trazar un lado.

De esta manera, se *salva* la variable de posición *Ex* con la que se realizaron los cálculos anteriores para polígonos con lados *n,* par o impar. Nótese que para simplificar pasos en el programa se utiliza una conjunción en las condiciones "n es par", e $i = \frac{n}{2} + 1$. Lo mismo para "n es impar" e $i = \frac{n-1}{2} + 1$.

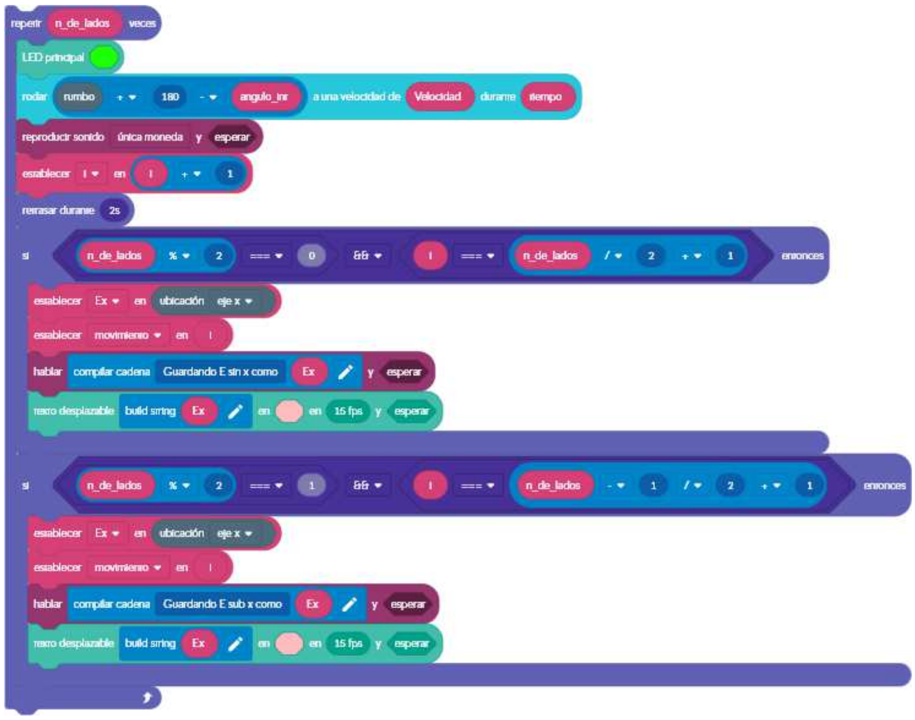


Figura 14. Programa en Sphero – segunda parte

Para hallar el *perímetro*, el Sphero dispone del sensor de "distancia" que arroja el valor total del recorrido al finalizar los movimientos. La longitud del lado l se obtiene dividiendo el valor del *perímetro* por *n*. Una vez obtenido el lado *l*, ya se pueden calcular los valores del radio y la apotema. Además, el área de un polígono regular es igual a la suma de las áreas de los *n* triángulos interiores: $Area = \frac{a.l}{2}.n$ siendo *a* la apotema, o $Area = \frac{a.Perimetro}{2}$ dado que $n.l = perimetro$.

La Figura 15, muestra la tercera, y última secuencia de comandos en la cual: se halla l con el método descripto. La apotema y el radio se calculan con una estructura de tipo *If-Then-Else*, para elegir el procedimiento dependiendo de si *n* es par o impar; y posteriormente, se obtiene el área del polígono. Para finalizar se imprimen todos los valores.

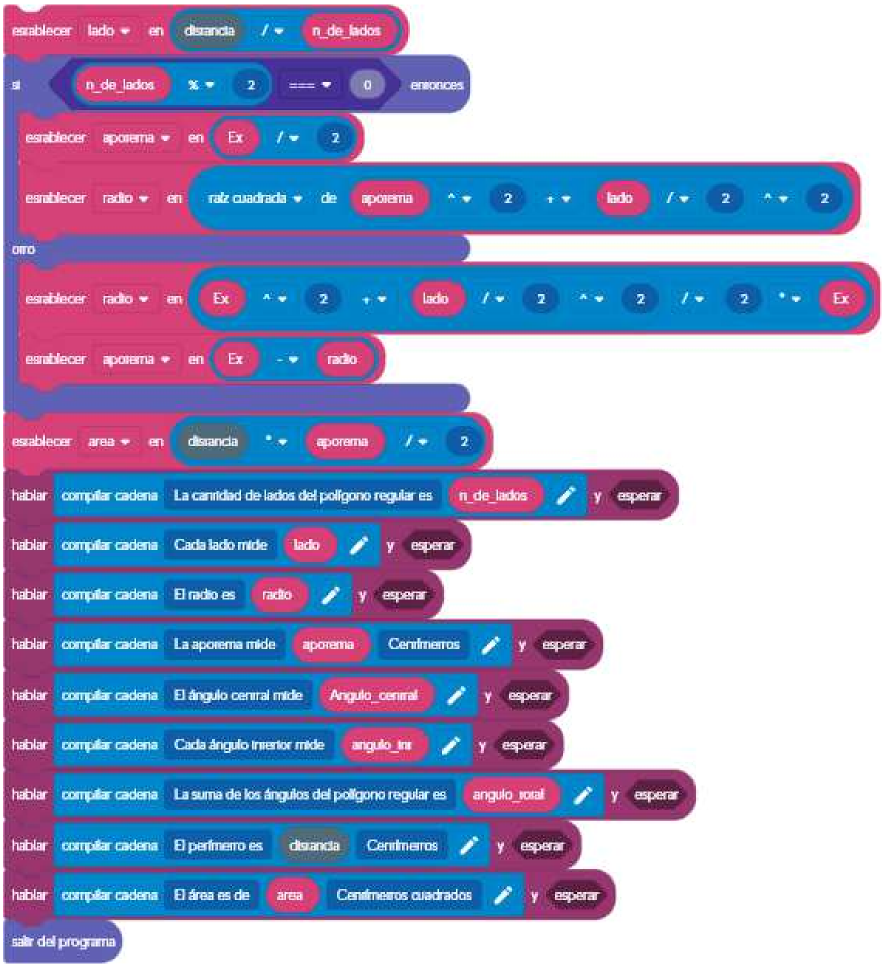


Figura 15. Programa en Sphero – tercera parte

6- DISCUSIÓN

El MPR y el REI aquí desarrollados muestran un posible recorrido que permite introducir la enseñanza de la programación en las clases de matemática de la ESA. La programación no tenía hasta hoy un espacio para ser enseñada en esta institución a pesar del interés de sus estudiantes. Tampoco existían en el estado del arte trabajos que reporten REI codisciplinares entre matemática y ciencias de la computación. En consecuencia, se trata de un movimiento a dos bandas que introduce un cambio de porte en la enseñanza de las matemáticas que viven en la ESA, mixturadas ahora con la programación y además en la OD, ya que la enseñanza es por investigación. Se evidencia una vez más, el potencial de la noción de praxeología como modelo de cualquier actividad humana regularmente realizada, en este caso programar un autómata.

Si bien el Ministerio de Educación argentino estableció la importancia de aprender a programar (CFE Nº 263/15), restricciones en el nivel de la Sociedad, muy alejadas de la escuela – como la recesión y los recortes del gasto público – han reducido la renovación tecnológica a un anhelo. Aunque existen propuestas para enseñar programación en el nivel primario y secundario, no hay trabajos relativos a la ESA y además esas propuestas requieren de “laboratorios” de informática, noción algo perimida y ausente en muchas instituciones.

En este caso, para desarrollar el REI en la ESA el uso de un dispositivo físico, un robot concreto y con un objetivo específico, es una mejor alternativa mucho mejor que la programación de objetos virtuales. Se prescinde del llamado “laboratorio de informática” pues solo se requiere de un robot Sphero, para que los estudiantes lo programen desde sus teléfonos celulares. Por su parte, la usabilidad del software y de la programación por bloques, presenta una ventaja sobre otros dispositivos y lenguajes de programación, en cuanto a la reducción del tiempo didáctico.

El robot Sphero es versátil y por sus características y precisión, podría ser usado para estudiar física o simplemente programación, en este caso, elegimos estudiar polígonos regulares, porque es parte del currículo de la ESA. Aquí la codisciplinariedad consiste en que la programación presta sentido a la matemática y viceversa. La matemática permite determinar qué es importante en el programa y cómo calcular los valores necesarios para que el programa funcione correctamente. Las estructuras de programación, los sensores y el sistema de representación en un eje cartesiano permiten a la matemática decidir los cálculos en relación a la paridad de lados del polígono.

En relación a los polígonos, y a la geometría en sí, es necesario cuestionar el lugar que ocupa en el programa de matemática y en el currículo realmente reconstruido, especialmente en la ESA donde hay menos tiempo disponible. Este saber no puede ocupar demasiado espacio en el currículo, pero tampoco puede reducirse a la memorización fórmulas. En el dispositivo, las praxeologías matemáticas vinculadas a los polígonos regulares y sus propiedades son estudiadas por su *utilidad formativa inherente* (Donvito, 2018) para lograr que un robot gire en la dirección correcta para trazar el polígono esperado. En este sentido, el REI permite revalorizar estos saberes que a esta altura ya se habrían "visitado" en otras instituciones, especialmente la geometría sintética que aquí resulta funcional, aunque tiende a extinguirse en el saber sabio. Se podría añadir la relación entre la geometría sintética y la analítica, habitualmente escindida, que resulta aquí complementaria. Esta interacción entre ambas geometrías permite que el recorrido se enriquezca, y permita al programa calcular la longitud de los lados, el radio, la apotema, el perímetro y el área de la figura trazada.

7-CONCLUSIÓN

En este trabajo se presentó y discutió el MPR provisorio relativo a un REI diseñado para estudiar de manera codisciplinar programación y matemática en la ESA. El estudio de praxeologías geométricas (sintéticas y analíticas), es motivado por el problema de desarrollar un programa para que un autómata trace un polígono regular y se estudian además en un pie de igualdad las estructuras básicas de programación. El MPR analiza las praxeologías vinculadas a las ciencias de la computación, disciplina que aún no había sido abordada en ningún REI y permite analizar la ecología y economía del dispositivo en el ámbito de la ESA.

REFERENCIAS

Chevallard, Y. (1999) El análisis de las prácticas docentes en la teoría antropológica de lo didáctico. *Recherches en Didactique des Mathématiques*, 19 (2), pp. 221-266.

Chevallard, Y. (2002). Approche anthropologique du rapport au savoir et didactique des mathématiques. *3es Journées d'étude franco-québécoises*. Université René-Descartes Paris 5.

Chevallard, Y. (2007). Passé et présent de la théorie anthropologique du didactique. En L. Ruiz-Higueras, A. Estepa, & F. Javier García (Éd.), *Sociedad, Escuela y Mathemáticas. Aportaciones de la Teoría Antropológica de la Didáctico* (pp. 705-746). Universidad de Jaén.

Chevallard, Y. (2013a). Enseñar Matemáticas en la Sociedad de Mañana: Alegato a Favor de un Contraparadigma Emergente. *Journal of Research in Mathematics Education, 2*(2), 161-182.

Chevallard, Y. (2013b). Analyses praxéologiques: esquisse d'un exemple. IUFM Toulouse, Francia. Recuperado desde http://yves.chevallard.free.fr.

Chevallard, Y. (2017). ¿Por qué enseñar matemáticas en secundaria? Una pregunta vital para los tiempos que se avecinan. *Gaceta de la Real Sociedad Matemática Española. 20*(1), 159-169.

Donvito, A. (2018). *Análisis macro-didáctico aplicado a la Educación Secundaria de Adultos: Praxeologías matemáticas y utilidad*. (Tesis doctoral). Facultad de Ciencias Exactas, UNCPBA. Tandil.

Donvito, A.; Otero, M. R. (2019a). Why do adults go to adult secondary school? *IKASTORRATZA e-Revista de Didáctica*, 23, 1-13.

Donvito, A.; Otero, M. R. (2019b). Utility of mathematical knowledge as a motivating factor in students' learning of Adult Secondary Education. *Educational Journal of the University of Patras UNESCO Chair*, *6*(2), 1-9.

Gaud, D. ; Minet, N. (2009). Parcours d'étude et de recherche en géométrie pour la classe de seconde. *Petit X*, 79, 49-71.

Habermas, J. (1987a). *Teoría de la acción comunicativa. Volumen 1: Racionalidad de la acción y racionalización social*. Madrid: Taurus.

Habermas, J. (1987b). *Teoría de la acción comunicativa. Volumen 2: Crítica de la razón funcionalista.* Madrid: Taurus.

Kim, S. (2015). *Les besoins mathématiques des Non-Mathématiciens quel destin institutionnel et social? Études d'écologie et d'économie didactiques des connaissances mathématiques*. (Thèse doctorale). Université Aix-Marseille.

Laudan, L. (1984). Science and Values. Berkeley. University of California.

Otero, M. R., Fanaro, M. A., Córica, A. R., Llanos, V. C., Sureda, P. & Parra, V. (2013). *La Teoría Antropológica de lo Didáctico en el aula de Matemática*. Buenos Aires: Dunken.

Parra, V; Otero, M. R. (2008). Praxeologías Didácticas en la Universidad: Un estudio de caso relativo al Límite y Continuidad de funciones. Revista Zetetiké, 17, 151-190.

Parra, V; Otero, M. R. (2018). Antecedentes de los Recorridos de Estudio e Investigación (REI): características y génesis. *REIEC 13*(2) 1-18.

Salgado, D. P. & Otero, M. R. (2020). Enseñanza por investigación en un curso de matemática de nivel universitario: los gestos didácticos esenciales. *Educ. Matem. Pesq., São Paulo, 22*(1), 532-557.

Quijano, M. T & Corica, A. R. (2017). Desarrollo de un Modelo Praxeológico de Referencia en Torno a Lugares Geométricos. *REDIMAT 6*(2), 192-220.

Recepción:09/10/2020 *Aceptación: 10/11/2020*

Angel Donvito

Profesor de Matemática. Licenciado en Educación Matemática. Doctor en Enseñanza de las Ciencias, mención Matemática por la Universidad Nacional del Centro de la Provincia de Buenos Aires (UNICEN). Becario post-doctoral en el Consejo Nacional de Investigaciones Científicas y Técnicas (CONICET). Ayudante diplomado ordinario en la Facultad de Ciencias Exactas (UNICEN). Miembro del Núcleo de Investigación en Educación en Ciencia y Tecnología (NIECyT).