

P.A.C.O, el robot dibujante

Introducción a la programación a través de la geometría plana en Educación Plástica, Visual y Audiovisual

Ramón Serrano Fernández
Facultad de Educación
Universidad de Salamanca

Salamanca, Castilla y León,
España
rserranof03@usal.es

RESUMEN

Este artículo presenta un proyecto de innovación docente en recursos, diseñado para introducir a estudiantes de primer curso de ESO en la programación y el arte generativo. A través de una plataforma digital y un lenguaje de programación simplificado, ambos denominados P.A.C.O (Programa Automático de Chapuzas por Ordenador), permite a los estudiantes crear arte generativo con gráficos de tortuga interactuando con una interfaz web simple. El desarrollo de competencias digitales y STEM es central, a través de la integración de la tecnología en el aula. El proyecto incluye una serie de actividades diseñadas para fomentar un aprendizaje activo y colaborativo entre los estudiantes, utilizando metodologías como Gamificación y Aprendizaje Basado en Problemas. Se detalla la implementación técnica de la plataforma P.A.C.O, incluyendo aspectos de su lenguaje de programación y la arquitectura de la plataforma web. Se presta atención especial a cómo la plataforma puede ser utilizada para enseñar conceptos básicos de geometría y programación, al mismo tiempo que fomenta un enfoque creativo y experimental hacia la educación artística y técnica. Este proyecto no solo busca mejorar las habilidades técnicas de los estudiantes, sino también introducirlos al arte generativo y al uso ético de recursos de IA generativa, usando los modelos ChatGPT y DALL-E, posicionándolos para utilizar la tecnología de manera creativa y responsable en el futuro. La integración de un *chatbot* basado en ChatGPT proporciona un recurso adicional para apoyar a los estudiantes en el aprendizaje y la exploración creativa.

PALABRAS CLAVE

Programación educativa, Inteligencia Artificial en educación, Plataforma web, Geometría, Innovación educativa en recursos.

ABSTRACT

This paper presents a resource-based teaching innovation project designed to introduce first-year ESO students to programming and generative art. Through a digital platform

and a simplified programming language, both called P.A.C.O. (Programa Automático de Chapuzas por Ordenador), it allows students to create generative art with turtle graphics by interacting with a simple web interface. The development of digital and STEM skills is central, through the integration of technology in the classroom. The project includes a series of activities designed to foster active and collaborative learning among students, using methodologies such as Gamification and Problem Based Learning. The technical implementation of the P.A.C.O. platform is detailed, including aspects of its programming language and the architecture of the web platform. Special attention is given to how the platform can be used to teach basic geometry and programming concepts, while fostering a creative and experimental approach to artistic and technical education. This project not only aims to improve students' technical skills, but also to introduce them to the generative art and ethical use of generative AI resources, using the ChatGPT and DALL-E models, positioning them to use technology creatively and responsibly in the future. The integration of a chatbot based on ChatGPT provides an additional resource to support students in learning.

KEYWORDS

Educational programming, Artificial Intelligence in education, Web platform, Geometry, Resources educational innovation.

1 Introducción

La presente propuesta constituye un proyecto de innovación docente en recursos, para el área de Educación Plástica, Visual y Audiovisual. Con este proyecto pretendemos iniciar al alumnado en la programación y el arte generativo. A través de un lenguaje y una plataforma muy sencillos, diseñados *ad hoc* para este fin, presentamos una secuenciación de actividades, a modo de muestra o ejemplo, que emplean estos recursos para implementar metodologías activas como la *Gamificación*, el *Aprendizaje basado en problemas*, o las competencias STEAM.

En cuanto a los conceptos más ligados a las matemáticas, esta propuesta se apoya principalmente en el

estudio de la **geometría euclidiana en el plano**, y busca estimular el **pensamiento y razonamiento lógico-matemático**. Se enmarca, como aquellos recursos en que se inspira, dentro del ámbito de la **programación educativa**, y está orientado principalmente hacia los aprendizajes **basados en problemas** y en **juegos**.

Hemos dado el nombre de P.A.C.O (Programa Automático de Chapuzas por Ordenador) al conjunto de plataforma y lenguaje, para presentarlo al alumnado como un pequeño robot dibujante, con cierto toque humorístico. El lenguaje de P.A.C.O es una variante muy reducida y básica del lenguaje de programación LOGO (1966). Uno de los distintivos de este lenguaje es que está traducido a un castellano muy coloquial, con el objetivo de resaltar este toque humorístico.

Habilitaremos también un *chatbot* dedicado a responder, ayudar, inspirar y dar ideas al alumnado, basado en el modelo de IA *ChatGPT*, que conoce específicamente el lenguaje, y es capaz de interpretarlo y modificar las imágenes resultantes del mismo, así como crear y proponer modificaciones a fragmentos de código.

La plataforma en la que se aloja P.A.C.O pretende, de la misma manera que el lenguaje, ser sencilla en el extremo: presentamos una sola página *web* estática, con un pequeño apartado de instrucciones y referencia, el intérprete para el lenguaje, y una clasificación de equipos.

Las actividades que presentamos en conjunto con la plataforma se pueden adaptar a una pequeña situación de aprendizaje:

- Introducción al lenguaje, en la que se puede prescindir de la plataforma
- Actividades en las que el alumnado intenta resolver retos de varios tipos relacionados con la geometría, el diseño, el arte generativo, y las matemáticas.
- Una propuesta para un proyecto o producto final: crear una pieza de arte generativo (en conjunto con el modelo de lenguaje *ChatGPT*).
- Un concurso final en el que los grupos ponen a prueba su destreza con P.A.C.O y la de sus pares.

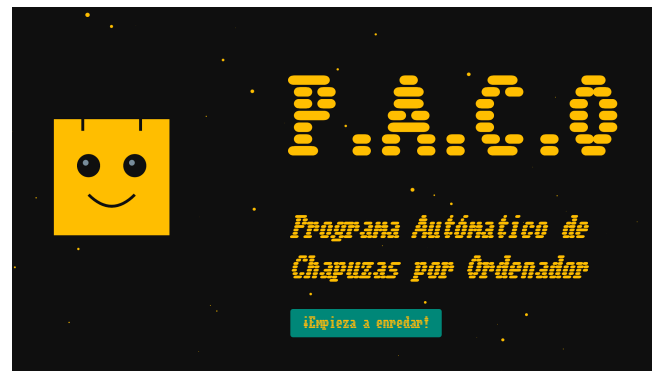


Figura 1: **Banner de bienvenida a la plataforma P.A.C.O**

2 Justificación

La vigente ley de educación (LOMLOE), que prescribe una educación competencial, recoge la *Competencia Digital* (CD), como una de las competencias clave que el alumnado debe alcanzar al término de la Educación Secundaria Obligatoria. En su preámbulo, la ley expone el desarrollo de esta competencia como un objetivo del sistema educativo:

Con el objetivo de que el sistema educativo adopte el lugar que le corresponde en el cambio digital, se incluye la atención al desarrollo de la competencia digital de los y las estudiantes de todas las etapas educativas, tanto a través de contenidos específicos como en una perspectiva transversal . . . (MECD, 2020, p. 122871)

En este instante de cambio digital, de saturación de imágenes e incipiente surgimiento de espacios y foros virtuales, que algunos autores como Fontcuberta, 2016, denominan *hipermodernidad*, creemos que la introducción de la creación de imágenes de manera algorítmica y la interacción con modelos de inteligencia artificial a la hora de llevar a cabo esta creación está a la orden del día.

En el ámbito de la Educación Plástica y Visual, nos parece tema de actualidad y de suma importancia educar en el uso responsable y ético de los recursos de inteligencia artificial que se ponen hoy en día a nuestra disposición. La UNESCO afirma en un artículo *web* sobre el Consenso de Beijing:

Promover la utilización equitativa e inclusiva de la IA, independientemente de cualquier discapacidad, estatus social o económico, origen étnico o cultural o situación geográfica, haciendo hincapié en la igualdad de género y garantizando la utilización ética, transparente y verificable de los datos educativos. (UNESCO, 2019)

Con este proyecto, pues, tratamos de educar al alumnado de manera transversal, desde el área de Educación Plástica, Visual y Audiovisual, tanto en la **competencia digital** como en las competencias **STEM** — y STEAM. Queremos aunar y presentar el uso de dos recursos automáticos y digitales muy diferentes para la producción de imágenes como son un lenguaje de programación y un modelo de inteligencia artificial. Esta comparativa puede resultar beneficiosa para el alumnado al favorecer la experimentación con la tecnología en un entorno controlado: descubrir diferentes acercamientos hacia la solución de un problema, cómo difiere el uso de diferentes recursos, qué puntos fuertes y flacos presentan, y qué beneficios y sinergias se obtienen al utilizarlos de manera conjunta.

Además, la plataforma P.A.C.O y las actividades que se plantean para su uso, dan un enfoque desenfadado y desde el área de Educación Plástica, Visual y Audiovisual a contenidos propios de las áreas en las que informes como PISA, (OCDE, 2023) reportan más carencias entre el alumnado de nuestro país — Matemáticas y Ciencias, ambas relacionadas con la competencia STEM. Estos contenidos son principalmente la **geometría** y el **razonamiento lógico**. Este enfoque, pese a que no es novedoso, pues contamos con precedentes como el lenguaje LOGO en 1966, o *Scratch* en 2012, puede resultar **atractivo** para el alumnado, dada la **caracterización** amigable de la plataforma y su componente **interactivo**.

Es, de hecho, el gran nivel de *feedback* e interacción que proporciona la plataforma y el lenguaje P.A.C.O, el motivo por el que creemos que son el recurso idóneo para impartir estos contenidos.

3 Objetivos

Nuestro objetivo principal al presentar este proyecto de innovación no es sino proponer un recurso didáctico que permita al alumnado, desde el área de Educación Plástica, Visual y Audiovisual, conocer algunas maneras de integrar el arte con las ciencias, la tecnología, y las matemáticas.

Para alcanzar este objetivo general, establecemos los siguientes objetivos específicos:

- Adaptar a las necesidades lingüísticas y simplificar un intérprete del lenguaje de programación LOGO.
- Desarrollar el pensamiento lógico-matemático, divergente, y lateral.
- Afianzar contenidos relacionados con el punto, la línea, el plano, y la geometría plana.
- Tener un primer contacto con el arte generativo.

4 Metodología

Para la realización de este proyecto, nos hemos apoyado en dos pilares fundamentales: una revisión bibliográfica sobre los conceptos y temas desarrollar, y un pequeño proceso empírico de experimentación.

Para la revisión bibliográfica hemos consultado en su mayoría artículos académicos acerca de la programación educativa: “History of Logo” (Solomon, C., et al., 2020), “Logo programming and problem solving” (Pea, R., D., 1987

De cara a la implementación de lenguajes de programación, hemos consultado el libro *Structure and Interpretation of Computer Programs* (Sussman, Sussman, Abelson, 1985). Sobre el papel social del juego, nuestro texto de referencia ha sido *Homo Ludens* (Huizinga, 1968).

Desarrollaremos también una comparación entre diferentes recursos de programación educativa como LOGO, *Processing*, o *Scratch* para determinar qué características y funcionalidades integrar en el lenguaje P.A.C.O.

Para la experimentación, hemos desarrollado tres sesiones en dos aulas diferentes de primero de ESO en el Colegio San Juan Bosco de Salamanca. De estas sesiones hemos recogido tanto resultados de trabajos con el recurso, como apuntes y observaciones de su desarrollo.

Sobre el uso y papel de la inteligencia artificial en el aula, además de los artículos mencionados, hemos extraído información de dos recomendaciones de la UNESCO: *Consenso de Beijing sobre la inteligencia artificial y la educación* (2019), y *Recomendación sobre la ética de la IA* (2022).

4.1 Análisis

En cuanto al análisis de la información recogida, realizamos una síntesis textual de los conceptos y puntos comunes de los artículos sobre la programación educativa.

Una tabla de funcionalidades nos servimos para comparar los diferentes recursos de programación educativa.

En cuanto a los resultados experimentales, nos serviremos de un diario de anotaciones sobre las sesiones, y de las propias producciones del alumnado, evaluando el grado de consecución de los objetivos de una tarea propuesta en la última de las sesiones.

5 Marco Teórico

La propuesta de innovación docente que presentamos tiene como principal inspiración el lenguaje de programación LOGO, aparecido en 1966, inspirado por el constructivismo de Piaget y las investigaciones en materia

de inteligencia artificial de Minsky (Solomon *et al.*, 2020). En concreto, adoptamos de este lenguaje de programación su vocación de accesibilidad y servir como recurso didáctico, y, desde el punto de vista técnico, los gráficos de tortuga, que LOGO ha popularizado. Otros recursos y referentes ligados a la programación educativa, e inspirados en parte por logo son *Scratch* (2012) o *Processing* (2001). La plataforma P.A.C.O hace uso en su implementación de la biblioteca de código *P5js* (www.p5js.org), otro recurso de programación educativa, aunque de carácter más técnico, y ligado a *Processing*.

Los aspectos más técnicos del proyecto se fundamentan en las investigaciones realizadas a mediados del siglo XX por Chomsky (1956), en materia de lenguajes formales, y por Turing (1936) en cuanto al concepto de computabilidad.

Del ámbito más cercano a las artes, bebemos en esta propuesta del **arte generativo**, y en especial de la **programación creativa**: una rama del arte que se sirve de algoritmos y procesos informáticos para la creación parcial o total de obras. Uno de los máximos exponentes de esta corriente artística es el movimiento conocido como *demoscene*, que usa como punto de encuentro y promoción foros de la red, como *pouët* (www.pouet.net). Este movimiento se caracteriza autoimponer límites y retos en cuanto a la programación de *demos* —la denominación de las piezas que crean—, generalmente en cuanto al lenguaje a usar, **el tamaño** del programa, o el sistema en el que debe funcionar.

En España el **Centro de cálculo de la Universidad de Madrid** fue en los años sesenta y setenta del siglo pasado un semillero de artistas relacionados con la informática. Contaba con pioneros en el ámbito de la **programación creativa** y el **arte generativo** como Eusebio Sempere y José María López Yturalde, cuyas obras sirven de inspiración para este proyecto y sus actividades.

UNESCO, 2019, anima a introducir recursos de IA en el ámbito educativo, sobre todo como un complemento a la docencia. A su vez, recomienda considerar ciertas cuestiones éticas respecto a su uso. En este proyecto, el modelo de IA *ChatGPT* (chat.openai.com) resulta esencial como *co-mentor*, en un entorno controlado y diseñado para ayudar al alumnado en las cuestiones relacionadas con las actividades propuestas, pero limitado, en la medida de lo posible, a la resolución íntegra de las mismas.

En lo relacionado con cómo se presenta esta propuesta al alumnado, usamos metodologías activas como la gamificación, y la competición. Dotar ciertos elementos de juego hace a las dinámicas propuestas en este proyecto potencialmente más atractivas para el alumnado. La función social y estructural del juego es crucial en el

proceso de aprendizaje en la niñez y adolescencia. Huizinga, 1968, nota que esto es especialmente importante principalmente en los mamíferos jóvenes — grupo en el que nos encontramos los humanos.

5.1 Antecedentes

Como antecedentes, hemos encontrado mayormente ejemplos de proyectos de innovación relacionados con la enseñanza de la programación. Hemos tomado referencias de dos de estos proyectos, ambos orientados al ámbito universitario: “Mejora de una Gymkhana de programación para reforzar el aprendizaje de Python” (Remeseiro López, B., *et al.*, 2019), y “Desarrollo de una página web para aprendizaje interactivo del lenguaje de programación R” (Romero Zaliz, R., *et al.*, 2011). En cuanto a proyectos que involucren tanto el arte como la tecnología, “Proyecto de arte y nuevas tecnologías”, del C.P.E.E. Andrés Muñoz Garde, en Pamplona, emplea la **robótica** y la **realidad aumentada** en los ámbitos de Ciencia, Arte y Tecnología.

El primer proyecto mencionado consiste en organizar una *gymkana* entre los estudiantes de una asignatura de introducción a la programación en la Universidad de Oviedo, durante el curso 2019/2020. Uno de los objetivos de este proyecto es afianzar los conceptos más importantes en el ámbito de la programación informática, y como incentivo, los resultados en las actividades contribuyen a la nota final de la asignatura.

El segundo proyecto tiene por objeto crear una plataforma *web* para el aprendizaje del lenguaje de programación R, en español y de manera autodidacta. Esta plataforma ha servido como apoyo a la docencia en la Universidad de Granada.

El último de los proyectos, que se dirige a la educación especial, es el único que integra las artes en su propuesta. Su principal objetivo es presentar las vocaciones científicas al alumnado mediante actividades que integran las artes como motor del discurso social, a través de la creación de robots y el uso de la realidad virtual y aumentada. Cabe destacar que este proyecto también es una propuesta inclusiva, que incluye tanto a centros de educación especial, como institutos de educación secundaria en Navarra.

6 Propuesta de innovación

Seguidamente presentamos nuestra propuesta de innovación docente en recursos, *P.A.C.O.*, *el robot dibujante*, que se puede desglosar en tres apartados:

- Lenguaje de programación P.A.C.O.
- Plataforma web P.A.C.O.

- Secuenciación de actividades para trabajar en la plataforma.

Tanto la secuenciación de actividades como una especificación informal del lenguaje se pueden encontrar en este documento. La plataforma se encuentra alojada en la red (www.ars-chromatica.art), y el código fuente de la misma, junto con el intérprete, están a disposición pública bajo licencia MIT, en el repositorio accesible a través de este enlace: <https://github.com/uri-nyx/PACO>.

6.1 Contextualización

El presente proyecto de innovación docente atiende a la realidad concreta de los cursos de primero de ESO del Colegio San Juan Bosco, situado en la avenida de Canalejas de la ciudad de Salamanca. Ambos cursos cuentan con treinta alumnos cada uno, y son bastante diversos en cuanto al nivel socioeconómico y la situación familiar. La mayor parte del alumnado se encuentra en la media: una situación familiar estructurada, de clase media, y con un rendimiento académico y de comportamiento normales. Sin embargo, distinguimos algunos casos especiales.

En 1ºA encontramos varios chicos de un nivel socioeconómico menor que el resto, al menos uno de ellos se encuentra bastante desatendido por su entorno familiar. Hay además dos personas con altas capacidades: una chica, que presenta características de socialización estándares, y un chico que parece tener dificultades para interactuar con sus pares, a falta de un diagnóstico concreto.

En 1ºB encontramos un alumnado mucho más homogéneo y sin gran disparidad. Tan solo una de las chicas parece presentar cierto retraso del desarrollo.

En cuanto a los recursos e instalaciones del centro, este cuenta con un aula de informática, y todas las aulas están equipadas con ordenador y proyector, aunque sin pizarra digital. Cabe destacar que el uso de los teléfonos móviles está bastante restringido de acuerdo con la política escolar: si el alumnado los trae a clase, deben ser guardados bajo llave en el aula de referencia, en sendos armarios dedicados a tal efecto.

6.2 Marco legal

Presentamos este proyecto de innovación docente para un centro que se encuentra en Salamanca, por lo que nos atenemos a la normativa de educación vigente a fecha de publicación en la Comunidad Autónoma de Castilla y León:

- Ley Orgánica 2/2006, de 3 de mayo, de Educación.

- Ley Orgánica 3/2020, de 29 de diciembre, por la que se modifica la Ley Orgánica 2/2006, de 3 de mayo, de Educación.
- Real Decreto 217/2022, de 29 de marzo, por el que se establece la ordenación y las enseñanzas mínimas de la Educación Secundaria Obligatoria.
- Orden EFP/279/2022, de 4 de abril, por la que se regulan la evaluación y la promoción.
- DECRETO 39/2022, de 29 de septiembre, por el que se establece la ordenación y el currículo de la educación secundaria obligatoria en la Comunidad de Castilla y León.

Extraemos del texto del DECRETO 39/2022, p. 49079, en lo concerniente a la asignatura de Educación Plástica, Visual y Audiovisual, los siguientes **objetivos**:

Desarrolla la creatividad y fomenta mediante el trabajo en equipo, actitudes de tolerancia, cooperación y solidaridad entre las personas y grupos.

La llegada de medios tecnológicos ha contribuido a enriquecer la disciplina democratizando la práctica artística, así como la recepción cultural, por este motivo desde la materia se desarrollan las competencias tecnológicas básicas facilitando la indispensable alfabetización del alumnado en este campo.

Este decreto también desarrolla la relación de esta materia con las dos **competencias clave** que creemos principales en el proyecto: **Competencia STEM** y **Competencia Digital**.

De la Competencia STEM el decreto, p. 49080, nota: “La geometría plana, la perspectiva y la representación objetiva de las formas permiten la utilización de las proporciones, dimensiones, relaciones, posiciones y transformaciones.”

De la Competencia digital: “Esta materia exige el uso creativo, crítico y seguro de las tecnologías de la información y la comunicación. . .” (p. 49080)

Desarrollamos la Competencia **STEM** a través de la introducción de un **lenguaje de programación** adaptado al aula, así como en el uso de este lenguaje para describir formas geométricas y objetos matemáticos simples. De la misma manera incorporamos la Competencia **Digital** en el empleo en el aula de este lenguaje de programación en una plataforma *web* específicamente dedicada para el proyecto.

El proyecto tratará los siguientes contenidos de los bloques B (p. 49090), y C (p. 49091):

- B2.** Elementos básicos del lenguaje visual: el punto, la línea y el plano. Posibilidades expresivas y comunicativas.
- B5.** La forma. Tipos y sus relaciones en el plano y en el espacio.
- C4.** Introducción a la geometría plana. Lugares geométricos. Trazados geométricos básicos.
- C5.** Figuras planas, Polígonos. Clasificación y construcción

6.3 Metodología de la propuesta

Para el planteamiento de este proyecto, hemos decidido centrarnos en los aspectos que lleven al alumnado a desarrollar las capacidades de **trabajo en equipo**, **pensamiento y razonamiento abstracto**, **lateral**, y **divergente**, así como la **solución de problemas**. Dadas las características principales del recurso que introducimos — un lenguaje de programación gráfica en su propia plataforma *web*—, la metodología principal y esencial de esta propuesta es la STEAM. Tenemos una intención clara de poner el conocimiento y herramientas de las matemáticas, la ingeniería y la tecnología al servicio del arte.

La metodología de **Aprendizaje basado en problemas** nos parece, también, muy importante, de cara a desarrollar los aspectos anteriormente citados. Además, creemos que las metodologías propias del **Pensamiento de juego** se adaptan bien al recurso — sobre todo la *gamificación*—, y le confieren dinamismo y atractivo al proyecto.

Los diferentes retos que propongamos al alumnado irán dirigidos a ejercitar e implementar las metodologías ABP. La fase de *concurso* tendrá por objeto motivar al grupo a través de la *motivación cognitivo-social*, e implementar aspectos de *gamificación*.

6.4 Temporalización

Recomendamos que esta propuesta se lleve a cabo durante al menos **doce sesiones**, lo que equivaldría, en la asignatura de Educación Plástica, Visual y Audiovisual, a aproximadamente un mes de trabajo.

Dado que aprender un lenguaje de programación, por sencillo que sea, supone un esfuerzo cognitivo considerable — más aún si es el primero que se aprende, como posiblemente suceda en este caso— creemos que es necesario introducirlo gradualmente. Esta introducción se puede realizar en unas **dos sesiones** dedicadas exclusivamente a este fin, pero recomendamos espaciarlo más en pequeñas actividades con anterioridad al comienzo propiamente dicho del proyecto.

Tras aprender el lenguaje, dedicaremos **cuatro sesiones** a familiarizar a los alumnos con la dinámica de trabajo grupal, la plataforma, y el proceso de puntuación de

los retos. Las siguientes **cuatro sesiones** las emplearemos en afianzar conceptos de geometría, y resolución de problemas relacionados con el arte y el diseño a través del pensamiento lógico-matemático. Tras esto, destinaremos **dos sesiones** a la realización de un trabajo final. La propuesta finaliza con otras **dos sesiones** de *concurso*, en la que el alumnado competirá por grupos.

Dinámica	Sesiones	Organización
Aprender Lenguaje	2, o actividades cortas	Individual
Retos sencillos	4	En grupo
Retos complicados	4	En grupo
Producto final	2	En grupo
Concurso	2	En grupo

Figura 2: Temporalización

6.5 Propuesta de innovación didáctica:

P.A.C.O el robot dibujante

El lenguaje P.A.C.O y su plataforma son un recurso digital interactivo, que permite al alumnado describir algoritmos extremadamente simples para dibujar figuras geométricas planas en la pantalla.

6.5.1 El lenguaje P.A.C.O

Dados el alcance y la complejidad del lenguaje, nos parece oportuno omitir en esta propuesta una especificación formal del lenguaje P.A.C.O. Sin embargo, creemos necesario describir su sintaxis, semántica, funcionalidad y las posibilidades que ofrece en profundidad, pues es la base de este proyecto.

Formalmente, este lenguaje es muy sencillo, tanto en sintaxis y semántica. Además, es realmente limitado en cuanto a poder computacional se refiere. Esta simplicidad y limitación del lenguaje están diseñadas *ad hoc*: buscamos una sencillez extrema, que nos permita enseñar y dominar rápidamente este lenguaje, y ponernos a dibujar y experimentar con él cuanto antes.

Consideramos que este lenguaje limitado es más que suficiente para presentar una iniciación a la programación, ya que permite implementar construcciones relativamente complejas, como bucles, y permite enseñar la base fundamental de la codificación de algoritmos.

6.5.1.1 Clasificación formal del lenguaje

El lenguaje P.A.C.O no es en realidad un lenguaje de programación universal. La razón principal es que este lenguaje no es *Turing completo*, que es una característica

de aquellos lenguajes y sistemas — computacionales o no—, que equivalen formalmente a una *máquina de Turing*.

El lenguaje que presentamos como recurso se encuentra en un estadio inferior al de una *máquina de Turing*, y equivale en capacidad computacional al modelo denominado *autómata de estado finito determinista* al carecer de **memoria** o la habilidad de especificar **saltos condicionales** en el programa. En la jerarquía de Chomsky, 1956, el lenguaje P.A.C.O equivale a un lenguaje *regular tipo-3*.

Este hecho supone que sea **imposible** codificar cierto tipo de algoritmos con P.A.C.O, como la parametrización de ecuaciones, dibujar espirales de manera algorítmica, curvas de *Bézier*, etc.

Debemos notar, sin embargo, que es teóricamente posible **aproximar** la mayoría de estas figuras, a partir de la descripción de los puntos que las componen.

6.5.1.2 Gramática y sintaxis formales del lenguaje

P.A.C.O es un lenguaje formal, lo que quiere decir que sigue unas reglas definidas cuya interpretación no es ambigua. Su sintaxis está inspirada principalmente en la usada en el lenguaje LOGO para manipular los *gráficos de tortuga*. Existen doce órdenes diferentes, entre las que diferenciamos tres tipos: aquellas que no toman parámetros, las que toman un número como parámetro, y la orden *repite*, que toma un número como parámetro, y una secuencia de órdenes.

Todas las órdenes tienen dos formas: la completa y la abreviada (e.g.: *repite* es la misma orden que *rep*). No hay ninguna diferencia semántica o gramatical entre la orden completa y abreviada.

La gramática formal del lenguaje, en formato EBNF es la que sigue:

```
programa = { orden } ;
orden = orden_sin_parametro
      | orden_con_parametro, numero
      | ( "repite" | "rep" ), numero, "[", { orden }, "]" ;
ordenes_con_parametro = "palante" | "pal"
                      | "patrás" | "pat"
                      | "derecha" | "dcha"
                      | "izquierda" | "izda"
                      | "grosor" | "gro"
                      | "color" | "col" ;
ordenes_sin_parametro = "esconde" | "esc"
                      | "enseña" | "ens"
                      | "punto" | "pt"
                      | "levanta" | "lev"
                      | "dibuja" | "dib" ;
numero = [ "-" ], { digito }, [ "." ], digito, { digito } ;
digito = "0" | "1" | "2" | "3" | "4" | "5" | "6" | "7" | "8" | "9" ;
```

6.5.1.3 Semántica y funcionalidad

El lenguaje P.A.C.O nos permite de manera nativa realizar tres tipos de acciones:

- *De movimiento*: estas acciones nos permiten mover y rotar el cursor por el espacio de dibujo.
- *De cambio de estado*: en todo momento, el intérprete se encuentra en un estado definido, que comprende los parámetros de **color**, **grosor del trazo**, **visibilidad del cursor**, y **si debe dibujar o no**.
- *De repetición*: el lenguaje posee una orden que nos permite repetir una secuencia de órdenes un número arbitrario de veces.

Esta relación describe el funcionamiento de cada instrucción concreta:

- *palante* (o *pal*) *n*: mueve el cursor hacia delante *n* píxeles.
- *patrás* (o *pat*) *n*: mueve el cursor hacia atrás *n* píxeles.
- *derecha* (o *dcha*) *n*: gira el cursor a la derecha *n* grados.
- *izquierda* (o *izda*) *n*: gira el cursor a la izquierda *n* grados.
- *grosor* (o *gro*) *n*: establece el grosor del trazo en *n* píxeles
- *color* (o *col*) *n*: establece el color del trazo al color con código *n*. El parámetro numérico hace referencia al código RGB del color. El intérprete de la plataforma permite especificar colores en formato hexadecimal (e.g.: *#aaff00*), o por su nombre en inglés, para colores que el navegador reconozca (depende del navegador utilizado).
- *esconde* (o *esc*): establece el cursor como invisible.
- *enseña* (o *ens*): establece el cursor como visible.
- *punto* (o *pt*): dibuja un punto cuyo grosor es el doble del establecido como grosor de trazo.
- *levanta* (o *lev*): en sucesivos movimientos, el cursor se moverá **sin dibujar**.
- *dibuja* (o *dib*): en sucesivos movimientos, el cursor se moverá **dibujando**. Es el comportamiento **por defecto** del intérprete de la plataforma.
- *repite* (o *rep*) *n* [*órdenes*]: repite un la secuencia de órdenes entre corchetes *n* número de veces.

6.5.2 La plataforma web P.A.C.O

La plataforma web P.A.C.O está alojada en la dirección www.ars-chromatica.art, y es una aplicación de una sola página. Incluye un intérprete del lenguaje P.A.C.O, una sección de ayuda y referencia, una sección de retos, y una sección de clasificación, habilitada para subir las puntuaciones de los diferentes equipos.

El código fuente de la plataforma se encuentra alojado en un repositorio del sitio GitHub (<https://github.com/uri-nyx/PACO>), con licencia MIT que permite su modificación sin restricciones. Este repositorio se puede descargar, y la página web funcionará sin necesidad de mayores modificaciones abriendo el archivo *index.html* en un navegador.

La interfaz de la plataforma ha sido diseñada completamente desde cero, con ánimo de conseguir una estética “robótica” y atractiva para el alumnado, pero a la vez sencilla y fácil de usar, sin adornos innecesarios.

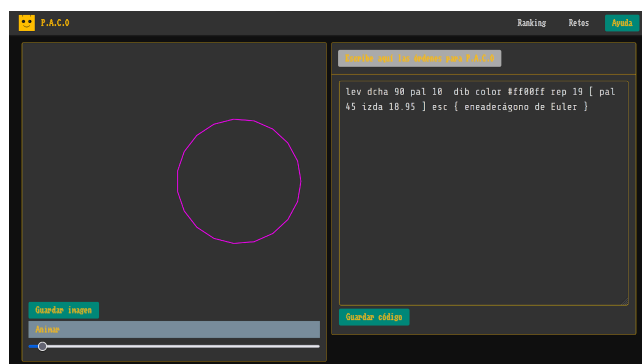


Figura 3: Intérprete del lenguaje P.A.C.O, con un programa que describe un eneadecágono.

Las fuentes utilizadas han sido obtenidas de Google Fonts (fonts.google.com): *Workbench*, de [Jens Kufílek](#); *Share Tech Mono*, de [Carrois Apostrophe](#); y *Silkscreen*, de [Jason Kottke](#). Estas fuentes se encuentran bajo la licencia *Open Font License* (www.openfontlicense.org).

La hoja de estilos principal de la plataforma está diseñada sobre el paquete *Picnic CSS* (www.picnicss.com), que se encuentra bajo licencia MIT. Los fondos animados de la versión de escritorio usan la biblioteca *particles.js* ([www.github.com/marcbruederlin/particles.js](https://github.com/marcbruederlin/particles.js)), también bajo licencia MIT.

El intérprete usa como motor la biblioteca *P5js* (www.p5js.org), bajo licencia *GNU LGPL v2.1* ([www.github.com/processing/p5.js/blob/main/license.txt](https://github.com/processing/p5.js/blob/main/license.txt)).

También, además de la referencia, integra un *chatbot* asistido por inteligencia artificial como *co-mentor*. Por el momento, hemos implementado esta funcionalidad con ChatGPT, y su uso requiere estar en posesión de una cuenta *premium* para el servicio.

Los gráficos usados en la plataforma (las imágenes de robots, y la estrella), son ilustraciones vectoriales en formato *svg* creadas por el asistente de inteligencia artificial ChatGPT.



Figura 4: Sección de ayuda de la plataforma.

6.5.3 Secuenciación de actividades propuesta

Debemos destacar que las actividades que a continuación presentamos no constituyen una situación de aprendizaje, y no están totalmente definidas. Son más bien una plantilla y perfilan de manera general como se puede emplear la plataforma P.A.C.O en el aula para impartir los contenidos curriculares citados en apartados anteriores, en conjunto con las metodologías activas ya mencionadas.

Agrupación: Proponemos dividir el grupo-clase en grupos pequeños de entre dos y cuatro personas para la realización de estas actividades: tanto para fomentar el trabajo en equipo, como para utilizar de manera eficiente los recursos del centro.

Entrega de tareas: Las entregas de diferentes tareas se realizarán o bien en papel, en las plantillas que se proporcionan, o bien a través de la plataforma digital de la que haga uso el centro.

El recuento de puntos otorgados a los equipos se debe publicar en la medida de lo posible en la plataforma P.A.C.O, de manera pública. En su defecto, se podría publicar en la plataforma del centro. Es importante por esto que los nombres de los grupos sean anónimos, y no mantengan relación con las identidades de sus integrantes.

6.5.3.1 Actividades para aprender el lenguaje P.A.C.O

Para aprender el lenguaje P.A.C.O proponemos dos actividades que se pueden llevar a cabo tanto en la plataforma interactiva, como en el aula.

1. Presentación: Primero presentaremos de manera amigable a P.A.C.O al alumnado como un robot dibujante. Introduciremos de manera gradual diferentes órdenes de movimiento, e intentaremos dibujar un cuadrado con ayuda

de la clase. Este dibujo lo podemos realizar en la pizarra, emulando los movimientos del robot. Tras el primer cuadrado, podemos pedir voluntarios para dibujar otras figuras. Cuando hayamos conseguido afianzar el uso de estas instrucciones básicas, lanzaremos la siguiente pregunta “¿Cómo puedo hacer un círculo?”. Buscamos aquí que el alumnado discorra, y descubra que necesita repetir una misma acción muchas veces. En este punto introduciremos la orden *repite*. Repasaremos la construcción de formas simples como cuadrados y triángulos introduciendo la orden *repite*.

2. Polígonos sencillos: Una vez el alumnado haya tenido la primera toma de contacto con paco, podemos introducir retos muy sencillos, como polígonos, y ejercicios de interpretación en los que se provee el programa y el alumnado debe dibujar su salida.

6.5.3.2 Retos sencillos para empezar a competir por grupos

Como se especifica en la temporalización propuesta, nos parece necesario emplear unas cuatro sesiones en proponer retos sencillos al alumnado, sobre todo de polígonos, figuras irregulares en las que especificamos distancias y ángulos, letras sencillas, etc. Estos retos servirán para terminar de entender el lenguaje, y para afianzar conocimientos de geometría, que podemos introducir en las mismas sesiones que estas actividades. En estas primeras sesiones, la mayoría de retos se pueden realizar en papel y en la pizarra, pero a medida que se van complicando, se torna necesario usar la plataforma interactiva.

1. Competición por grupos: Formamos los grupos, les pedimos que se registren en un formulario — ver anexo A—, y les proponemos su primer reto: dibujar un hexágono regular. Especificamos los criterios para otorgar puntuaciones — recomendamos un punto por rapidez, un punto por solución correcta, y uno a la solución con menor número de órdenes. El primer grupo en resolver el reto puede introducir su programa en el ordenador de clase para comprobar si es correcto. Podemos repetir con varios retos, tratando de dar oportunidades a todos los grupos de intentar “resolver”, y resolviendo las dudas que pudieran surgir.

6.5.3.3 Retos complicados para ejercitar el pensamiento lógico-matemático

Una vez el lenguaje esté totalmente afianzado, podemos pasar a proponer retos de más enjundia, que pueden ocupar la sesión entera tras una pequeña explicación de los conceptos matemáticos o geométricos subyacentes. Estos retos pueden ser más abstractos que solamente dibujar: describir cómo se completaría un reto

sin llegar a escribir el código, solo explicar el procedimiento, conseguir reducir el número de órdenes de un programa sin cambiar su salida, determinar si una figura propuesta podría ser dibujada con P.A.C.O o no, etc. Además, se pueden proponer retos de dibujo de figuras más complejas, como círculos concéntricos, espirales, estrellas o palabras enteras.

En el caso de que propongamos retos abstractos, debemos estar seguros de que tienen solución, o de que el alumnado entiende ciertos conceptos abstractos de la teoría de la computación y las matemáticas, como que algunos problemas **no son decidibles**, o **no son computables**, así como las limitaciones del propio lenguaje. Algunos problemas, como determinar si un programa es óptimo para cierta figura, requieren también de pruebas matemáticas complejas que se deberían evitar.

1: Dibujar el nombre del grupo.

2: Dibujar una espiral.

3: Dibujar a P.A.C.O o una estrella. En el caso de escoger y completar la estrella, se otorgará un bonus de puntos al grupo.

4: Partir de un programa ineficiente dado para una letra del alfabeto, un polígono o figura regular como un eneadeágono, o una estrella, y pedir que consigan reducir el número de órdenes empleadas.

6.5.3.4 Producto o trabajo final propuesto

Para introducir los conceptos de arte generativo, e implementar los contenidos curriculares que tratan sobre la creación de obras artísticas, proponemos como trabajo final la creación de una obra de arte generativo. Para esta obra podemos mostrar obras de la *demoscene*, del Centro de Cálculo de la Universidad de Madrid, o de otros artistas notables de *net-art* o *media-art* como inspiración. Estableceremos una serie de criterios de cara a la evaluación de las obras, estipulando cuántos elementos y con qué nivel de complejidad deben contar. Permitiremos libertad en todos los demás ámbitos, y requeriremos al alumnado a que se apoye en herramientas de IA como ChatGPT y DALL-E, tanto para conseguir ideas e inspiración como para obtener piezas generativas basadas en su entrada. Este proyecto es recomendable realizarlo íntegramente con acceso a un ordenador o dispositivo por grupo.

6.5.3.5 Concurso final

Como colofón de este proyecto, proponemos realizar un concurso entre los grupos. En este concurso propondremos retos parecidos a los que hemos realizado ya, con el añadido de que los grupos a la cabeza de la clasificación también deben proponer desafíos a sus pares. Con esta

actividad buscamos dejar un buen sabor de boca, y dinamizar el grupo con un ambiente de juego competitivo. También buscamos que tengan protagonismo los grupos a la cola de la clasificación, en la que es posible que haya surgido una brecha entre grupos con más puntuación y menos a lo largo del proyecto. Es posible también dinamizar este concurso introduciendo retos propuestos por el *chatbot* de IA experto en el lenguaje.

Conclusiones

Una de las primeras conclusiones que extraemos es que reducir a **tres** las sesiones de la propuesta acrecienta considerablemente la disparidad entre los grupos en cuanto al grado de manejo del lenguaje. En uno de los grupos contamos con al menos dos personas con altas capacidades intelectuales, cuyos grupos se han situado rápidamente a la cabeza de la clasificación.

Esta reducción en el número de sesiones se ha debido principalmente a la programación curricular ya establecida para el curso en el que hemos planteado la propuesta.

Debemos destacar también que las interacciones del alumnado con el *chatbot* de IA y el modelo *ChatGPT* en general, no han sido posibles de manera general, debido a la política del centro respecto al uso de estos recursos, que prohíbe su uso en la red escolar mediante un *firewall*. Algunos grupos de alumnos que así lo han deseado, han experimentado conversaciones breves con este *chatbot* a través de nuestro dispositivo personal.

Otras eventualidades logísticas se han presentado en el centro, como la imposibilidad de acceder a la sala de informática en todas las sesiones, por lo que hemos tenido que adaptar los retos para ser resueltos en papel, y luego probados por solo algunos grupos en el ordenador del aula. Esto incrementa la dificultad de retos creativos, o en los que pretendemos reproducir una figura, pues el alumnado carece de las referencias exactas que seguirá el intérprete a la hora de describir ángulos o distancias.

Pese a todo, la propuesta ha sido bien acogida, y la mayoría del alumnado parece haber comprendido al menos las partes más básicas del lenguaje y la construcción de figuras sencillas como polígonos o círculos. En cuanto a conceptos de lenguaje visual, queda patente que es necesaria la interacción y experimentación con el lenguaje

para alcanzar a comprenderlos mediante el mismo. También parecen haber quedado aprendidos ciertos conceptos de programación, como la secuenciación de acciones, descripción de programas, o la repetición de órdenes.

El alumnado presenta una respuesta muy positiva ante la propuesta, y sobre todo está muy interesado en usar la plataforma en su tiempo libre. La personificación y caracterización del lenguaje y la plataforma P.A.C.O como *Paco*, *el robot de clase que dibuja* ha sido un rotundo éxito.

REFERENCIAS

Chomsky N. (1956). Three models for the description of language. *IRE Transactions on Information Theory* 2(3): 113–124.

C.P.E.E Andrés Muñoz Garde. (s.f). *Proyecto de arte y nuevas tecnologías*. <https://andresmunozgarde.educacion.navarra.es/web/proyecto-de-arte-y-nuevas-tecnologias/>

Fontcuberta, J. (2017). La furia de las imágenes. Notas sobre la postfotografía. Gustavo Gili.

Huizinga, J. (1968). *Homo ludens: A study of the play element in culture* (17th ed.). Beacon Press.

OCDE. (2023). *PISA 2022 Results: Factsheets*. <https://www.oecd.org/publication/pisa-2022-results/country-notes/spain-f1a3afc1#chapter-d1e11>

Remeseiro López, B., Cebrían Márquez, G., De la Cal Marín, E. A., Díaz Honrubia, A. J., García Fanjul, J., Rico Pachón, N., Villar Flecha, J. R., y Suárez-Otero González, P. (2019). *Mejora de una Gymkhana de programación para reforzar el aprendizaje de Python*. Universidad de Oviedo.

UNESCO. (2019, 3 de julio). *La UNESCO ha publicado el primer consenso sobre la inteligencia artificial y la educación*. <https://www.unesco.org/es/articulos/la-unesco-ha-publicado-el-primer-con-senso-sobre-la-inteligencia-artificial-y-la-educacion>

Solomon, C., Harvey, B., Kahn, K., Lieberman, H., Miller, M. L., Minsky, M., y Silverman, B. (2020). History of logo. *Proceedings of the ACM on Programming Languages*, 4(HOPL), 1-66

Turing, A. M. (1936). On computable numbers, with an application to the Entscheidungsproblem. *J. of Math*, 58(345-363), 5.

Zaliz, R. C. R., Fernández, J. A., Robles, V. A. D., Castro, P. V., del Jesus, M. J., Cortes, J., y del Val Muñoz, M. C. (2011). Desarrollo de una página web para el aprendizaje interactivo del lenguaje de programación r. En *I Congreso Internacional de Innovación Docente. CIID: Cartagena 6, 7 y 8 de julio de 2011* (pp. 675-684).

BIBLIOGRAFÍA


Abelson, H., Sussman, G. J., & Sussman, J. (1996). *Structure and Interpretation of Computer Programs* (2nd ed.). The MIT Press.

López, A., y Munarriz J. (2021). *El Centro de Cálculo de la Universidad de Madrid (1968-1973): ciencia, arte y creación computacional*. Ediciones Complutense.

UNESCO. (2019). Consenso de Beijing sobre Inteligencia Artificial y Educación. <https://unesdoc.unesco.org/ark:/48223/pf0000368303>

UNESCO. (2021). *Recommendation on the Ethics of Artificial Intelligence*. <https://unesdoc.unesco.org/ark:/48223/pf0000381137>

Anexo A: formularios y plantillas propuestos



P.A.C.O.
Programa automático de
chapas por ordenador

**REGISTRO
DE EQUIPOS**

GRUPO Y
GRUPO

FECHA

HOMBRE DEL
CENTRO

HOMBRE DEL
EQUIPO


INTERVENIENTE
DEL EQUIPO

LEÑA DEL
EQUIPO

DIBUJO DEL
EQUIPO

PD0: LOS INTEGRANTES
DEL EQUIPO

PD0. PROFE:



P.A.C.O.
Programa automático de
chapas por ordenador

**INFORME
DE RETO**

GRUPO Y
GRUPO

FECHA

HOMBRE DEL
CENTRO

HOMBRE DEL
EQUIPO

TIPO DE RETO

PUNTUACIÓN

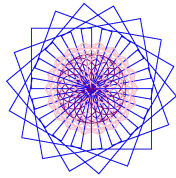
REQUISITOS Y
CONTORNOS DE
PUNTUACIÓN

CÓDIGO

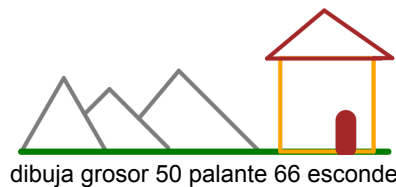
IMAGEN

Estas plantillas pueden ser útiles para la recogida de información sobre los equipos, o para proponer retos como tareas en papel.

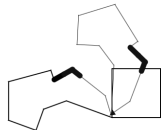
Anexo B: piezas del alumnado



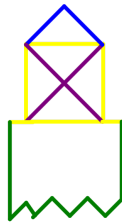
repite 360 [palante] izquierda 100 color pink repite 360 [palante 100 izquierda 80 palante 1] color blue repite 360 [palante 200 derecha 90] color purple repite 360 [palante 1 derecha 1]



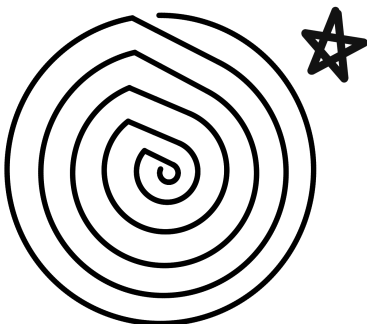
dibuja grosor 50 palante 66 esconde



rep 4 [palante 100 izda 90] dcha 200 palante 100 izda 40 palante 50 izda 50 palante 46 dcha 90 palante 60 dcha 70 palante 79 dcha 75 palante 90 dcha 86 palante 20 izda 100 grosor 11 palante 23 (col yellow) palante 20 dcha 67 palante 20 grosor 1 palante 67 dcha 35 palante 40 dcha 200 palante 100 izda 40 palante 50 izda 50 palante 46 dcha 90 palante 60 dcha 70 palante 79 dcha 75 palante 90 dcha 86 palante 20 izda 100 grosor 11 palante 23 (col yellow) palante 20 dcha 67 palante 20 grosor 1 palante 67 dcha 35 palante 40



grosor 10 color yellow palante 200 izquierda 90 palante 200 izquierda 90 palante 200 izquierda 90 palante 200 color purple izquierda 135 palante 280 izquierda 90 color blue palante 140 izquierda 90 palante 140 color purple izquierda 90 palante 280 color yellow izquierda 45 palante 50 color green derecha 90 palante 200 color green derecha 45 palante 60 derecha 90 palante 65 izquierda 90 palante 65 derecha 90 palante 65 izquierda 90 palante 70 levanta color yellow derecha 135 palante 245 izquierda 90 dibuja palante 60 izquierda 90 color green palante 250 izquierda 135 palante 60 derecha 100 palante 40 esconde



levanta izquierda 90 palante 270 derecha 90 dibuja grosor 10 repite 350 [palante 5 derecha 1 groso] derecha 41 palante 2 izquierda 4 palante 48 esconde palante 130 repite 320 [palante 4 derecha 1] derecha 45 palante 2 izquierda 4 palante 51 esconde palante 100 repite 310 [palante 3 derecha 1] derecha 50 palante 2 izquierda 5 palante 51 esconde palante 68 repite 300 [palante 2 derecha 1] derecha 65 palante 2 izquierda 5 palante 51 esconde palante 50 repite 290 [palante 1 derecha 1] derecha 78 palante 2 izquierda 5 palante 57 esconde palante 0 repite 270 [palante 0.3 derecha 1] levanta grosor 15 color yellow enseña derecha 25 palante 420 izquierda 49 palante 30 derecha 200 dibuja palante 120 izquierda 140 palante 110 izquierda 143 palante 110 izquierda 140 palante 110 izquierda 145 palante 120 esconde