

EL ACTO DE ESCRITURA: LA PROGRAMACIÓN ORIENTADA A OBJETOS COMO PRODUCTORA DE CONOCIMIENTO DESDE LA PRÁCTICA ARTÍSTICA

THE WRITING ACT: OBJECT-ORIENTED PROGRAMMING AS KNOWLEDGE PRODUCER FROM ARTISTIC PRACTICE

YUNUEN ROSAS ORTIZ

Universidad Autónoma del Estado de México, México
<https://orcid.org/0000-0002-5486-4037>
yun.lagotadelryo@gmail.com

Recepción: 10 de septiembre de 2020

Aprobación: 14 de enero de 2021

RESUMEN

“El acto de escribir”, no como un problema filosófico sino como un “estar haciendo”, deja entrever un método de investigación que puede tener lugar dentro de las prácticas artísticas con programación, siendo un proceso reflexivo a través del texto relacionado con la escritura de algoritmos y la Programación Orientada a Objetos (POO). La escritura de sistemas regulativos, como los propuestos en la producción de arte generativo o la literatura electrónica, es consecuencia de un análisis interpretativo del mundo, por lo que un programador, como un traductor del mundo para las máquinas, las propone como agentes colaborativos de la producción de conocimiento humano en tanto la construcción de estos sistemas se relacionan con el proceso de pensamiento de su programador. La POO como un “acto de escribir” desde la práctica artística, abre la posibilidad de espacios de reflexión y conocimiento.

Palabras clave: máquina, arte generativo, caja negra, algoritmos, método de investigación.

ABSTRACT

The writing act, not as a philosophical problem but something that is doing, suggests a research method that can take place within artistic practices with programming, being a reflective process through the text related to the writing of algorithms and the Object-Oriented Programming (OOP). The writing of regulative systems, such as those proposed in the production of generative art or electronic literature, is the consequence of an interpretive analysis of the world, so that a programmer, like a translator of the world for machines, proposes them as collaborative agents of the production of human knowledge while the construction of these systems is related to the thought process of its programmer. OOP as an act of writing from artistic practice, opens the possibility of spaces for reflection and knowledge.

Key words: machine, generative art, black box, algorithms, research method

INTRODUCCIÓN

Existen varias formas de estructurar un programa informático, a estas estructuras se les conoce como “paradigmas de programación”.¹ Hay diversos tipos de paradigmas dependiendo de cómo estos ordenan la información que proporcionan a la máquina que ejecuta un programa. Uno de los más importantes en nuestros tiempos es la Programación Orientada a Objetos (POO), que surge en Noruega en 1967. Desde entonces ha sido implementada en diversos lenguajes de programación como el lenguaje Small Talk, C, C++, JAVA y JAVA Script que permiten experimentar con los datos que maneja un programa, solucionando uno de los principales problemas del desarrollo de *software* a principios de los años 70 del siglo pasado, en los que había que priorizar la planificación del programa, debiendo tener claras las necesidades y dificultades a las que se enfrentaría el programador y el usuario antes de escribir la primera línea del *software*, cosa que en la práctica es imposible.

La POO está compuesta por estructuras de autorreferencia, siguiendo cuatro pilares:² la abstracción, la herencia, el polimorfismo y el encapsulamiento. Estas estructuras se convierten en una forma de reflejo del mundo fuera de la pantalla que corresponden a procesos de traducción desde el pensamiento humano al entendimiento de las máquinas a través de la escritura de algoritmos que se encuentran ocultos en “la caja negra”, concepto retomado desde la cibernética como un lugar inaccesible para el usuario de un objeto tecnológico y que esconde los procesos de pensamiento y entendimiento del mundo de quien programó a una máquina, dejando visible únicamente la entrada y salida de un sistema.

En este trabajo se indaga a la POO como un proceso de pensamiento escrito que tiene la posibilidad de construir y alterar mundos desde “el acto de escribir”, con la intención de acercarnos a “la caja negra” de las máquinas desde las prácticas de programación en las artes, invitando a reflexionar sobre las prácticas artísticas tecnológicas como un medio con la potencia de ensamblar discursos filosóficos, sociales, culturales y políticos.

Aquí, se piensa al “acto de escribir”, no como un problema filosófico sino como un “estar haciendo”, que deja entrever un método³ de investigación y producción que propone simultaneidad en su escritura y lectura abriendo espacios de conocimiento dentro de la práctica artística. La programación se relaciona directa pero no únicamente con los actos de escritura debido a su potencial como forma de pensamiento, por lo que no profundizará mucho en el funcionamiento de la POO sino en su relación con el acto de escritura como generadora de conocimiento, es por esto que es importante conocer algunas características de este paradigma.

¹Véase J. Rodríguez (2003). *Introducción a la programación. Teoría y práctica*, pp. 4-6.

²Estos pilares serán desarrollados más adelante.

³Véase Juan Carlos Arias (2020: 8), quien propone al método de investigación en artes como una dinámica susceptible de repeticiones que produce una distancia sobre el objeto de investigación.

¿CÓMO FUNCIONA LA PROGRAMACIÓN ORIENTADA A OBJETOS?

La POO es una estructura escrita compuesta por objetos y clases (INFOTEC, n.d.: 8-9). Las clases son un elemento abstracto, contienen objetos de un mismo tipo definiendo atributos y métodos comunes entre ellos, siendo los atributos los datos que un objeto de la clase puede usar y los métodos los procedimientos que actúan sobre los mismos. Pongamos por ejemplo la clase animal (Imagen 1).

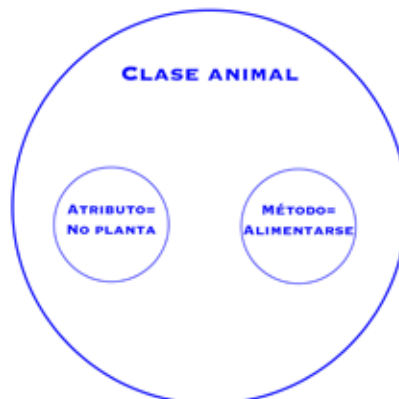


Imagen 1. Clase animal. Elaboración propia.

En este momento la clase animal cuenta solo con un atributo —no ser planta— y un método —alimentarse—. Al no tener un objeto al que dar sus características no puede comunicarse con el exterior.

Los objetos son bloques de texto con los que podremos comunicarnos al momento de ejecutar un programa, son el cuerpo de la clase. Sin los objetos, los atributos y métodos de una clase, no surten ningún efecto ya que no hay nada a que atribuírselos (Imagen 2).

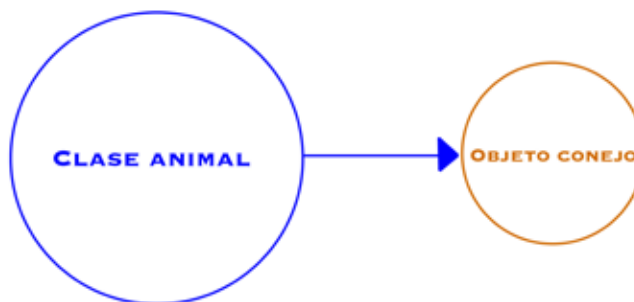


Imagen 2. Clase animal con objeto conejo. Elaboración propia.

Aquí se ha agregado el objeto conejo que pertenece a la clase animal. El objeto conejo puede interactuar con su entorno según las características que le permite su clase.

La POO tiene cuatro pilares que son la base de su lógica, y que permiten que su escritura tienda hacia la creación y experimentación:

1. Herencia. Este pilar propone crear clases, partiendo de otras que ya existen, compartiendo atributos y métodos entre ellas, a la vez que se añaden nuevas características y funciones. De esta forma, se pueden resolver problemas más complejos sin la necesidad de partir de cero (Imagen 3).

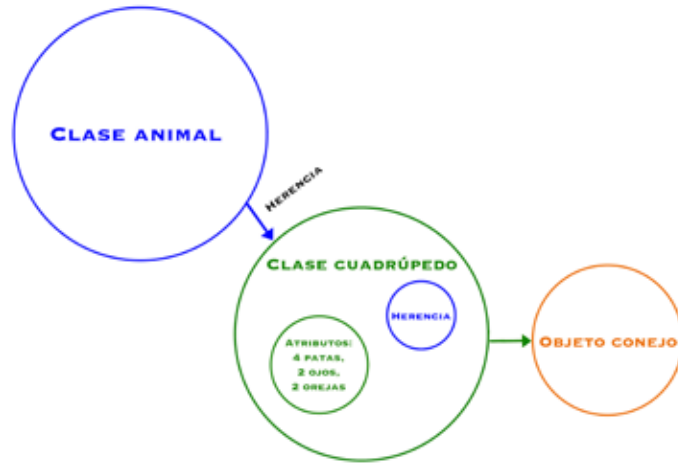


Imagen 3. Herencia. Elaboración propia.

2. Polimorfismo. Que propone que un objeto puede compartir características de dos o más clases. De esta manera, un objeto puede obtener las posibilidades, variables y métodos desde clases diferentes, lo que le permite acceder a más información construyendo sistemas más complejos (Imagen 4).

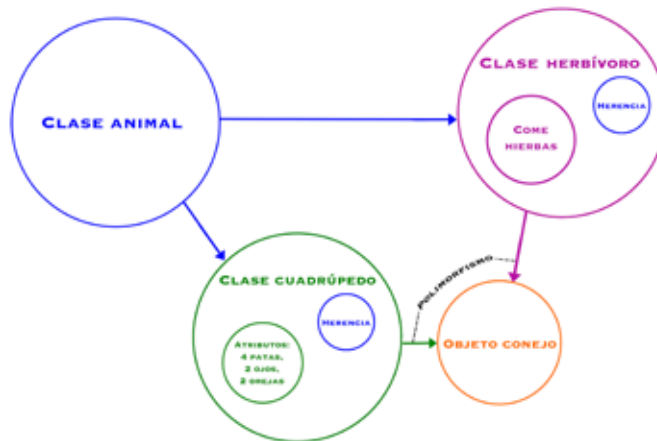


Imagen 4. Polimorfismo. Elaboración propia.

3. Encapsulamiento. Restringe el acceso de los objetos a atributos y métodos específicos de las clases. Así, algunos atributos y métodos son de uso “público” para que todos los objetos de un programa puedan acceder a él, otros son de uso “protegido” a los que solo pueden acceder los objetos de un mismo módulo⁴ y otros de uso “privado” para una sola clase (Imagen 5).

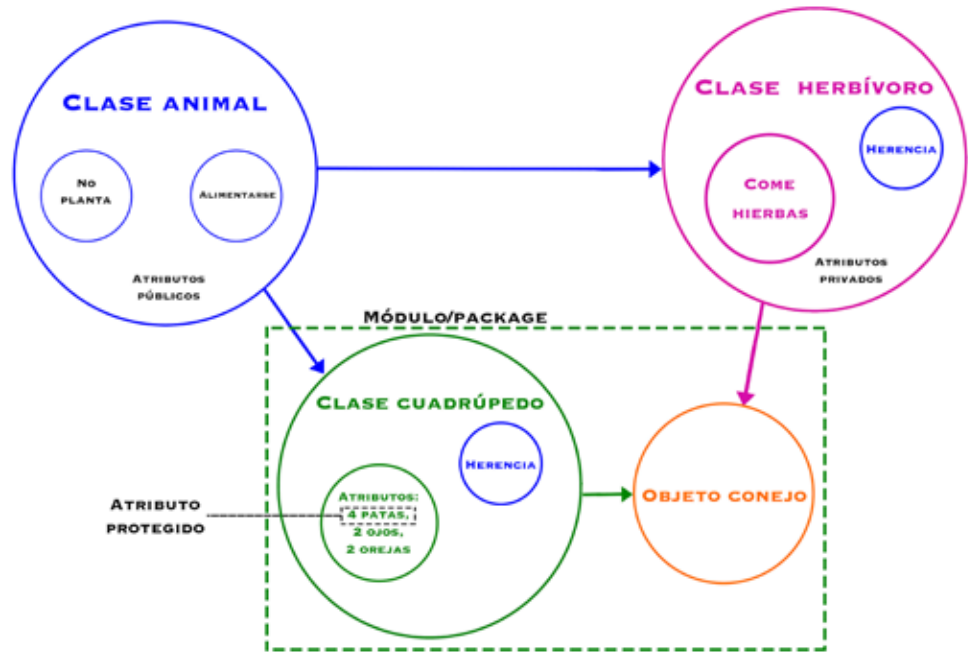


Imagen 5. Encapsulamiento. Elaboración propia.

Estos ejemplos no tienen ninguna funcionalidad, pero nos ayudan a entender la estructura de la POO. Este paradigma propone un estilo de escribir por bloques de texto, que hace autorreferencia usando sus clases y sus objetos, ayudando a los programadores a construir un sistema para conseguir algún resultado. ¿Cómo se plantea el camino a un resultado usando la POO?

La POO está escrita con base en una serie de instrucciones llamada “algoritmo”, y los algoritmos derivan del cuarto pilar:

4. Abstracción. Este pilar no tiene que ver con la estructura del texto, sino con la forma en la que se piensa. La abstracción permite pasar de las ideas al texto.

⁴ Un módulo o *package* es un contenedor de código. Pueden albergar varias clases, objetos y otros *package*, adoptando una jerarquía de contenedores en ese caso.

ALGORITMOS: UN MUNDO TRADUCIDO

Los algoritmos son secuencias de pasos jerárquicos finitos que permiten hallar la solución a algún problema. Un algoritmo es una receta de cocina, en la que se separa cada tarea en acciones pequeñas y concretas para hacer una secuencia de instrucciones específicas repetitivas y así llegar a un resultado esperado. También, aplicamos algoritmos al jugar juegos de mesa, al seguir las instrucciones para armar algún juguete o al transitar por las ciudades en automóvil.

La POO construye sistemas computacionales que permiten llegar a un resultado, otorgándoles autonomía a través de la construcción de algoritmos. Es importante pensar al sistema como un “molde interno” (Solaas, 2010), en el cual la información que entra se adhiere a un orden específico con el que se construye un objeto nuevo. Un ejemplo puede ser la cantidad de jugadores de un juego de mesa: Si pensamos que cada jugador es un objeto de la clase “serpientes_y_escaleras”,⁵ el número de objetos que pertenecen a la clase no afecta a la clase, se incorporan y se organizan en el sistema, ejecutando los atributos y métodos que su clase les permite; es decir, sin importar el número de jugadores, las reglas del juego “serpientes y escaleras” no cambian. Ese “molde interno” está determinado por los algoritmos. ¿Cómo se escriben los algoritmos?

Los algoritmos son construidos desde el análisis subjetivo del mundo del escritor que programa a una máquina: el programador. ¿Qué hace un programador? Un programador escribe, se enfrenta a un vacío lleno de posibilidades desde el texto, inventando sistemas de reglas partiendo de su propia relación con el mundo, siguiendo procesos de interpretación, abstracción y experimentación con el lenguaje computacional que le permite comunicarse con las máquinas.

Leonardo Solaas (2019), dice que el lenguaje utilizado por los programadores para comunicarse con las máquinas contiene una “función regulativa”, cuyo objetivo es establecer restricciones —más no órdenes, porque estas no permitirían la adhesión de nuevas posibilidades—, imaginando un problema para luego pensar su solución en cada paso, lo que lo vuelve diferente a cualquier otra función del lenguaje humano ya que construye sistemas de reglas a través de un texto que se encuentra a la mitad entre el inglés y las matemáticas, en un formato visualmente similar a las estrofas de un poema. Desde estos sistemas regulativos, las máquinas tienen la posibilidad de tener contacto con el mundo fuera de ellas, se convierten en agentes colaborativos de la producción de conocimiento humano en tanto la construcción de estos sistemas se relacionan con el proceso de pensamiento de su programador al buscar la autonomía parcial o total de la máquina en cuestión porque la

5 Juego de mesa de origen mexicano en el que un grupo de jugadores tira dados y avanza casillas hasta llegar al final del tablero.

escritura de sistemas regulativos es consecuencia de un análisis interpretativo del mundo partiendo del planteamiento de un problema o cuestionamiento.

Es por esta acción interpretativa que la forma en la que las máquinas funcionan depende totalmente de quién las ha programado; un programador es un traductor del mundo para las máquinas, porque desde esa posición de intérprete ocurren procesos de experimentación que derivan en la construcción de algoritmos en los que se especifica un contenido de ideas y decisiones producto de la posición y deseo de quien escribe. Por tanto, el entendimiento que una máquina tiene del mundo no es imparcial ya que “[...] no existe el susodicho traductor oculto” (Canales, 2018: 73). La posición ante el mundo que adopta una máquina está sujeta a la posición sociocultural y política de quien la programó.

La escritura en POO, al permitir relacionar bloques de texto —que funcionan como contenedores de ideas— como relacionaríamos los objetos del mundo tangible (INFOTEC, n.d.): los mismos atributos para objetos de una misma clase o tipo, nos da la posibilidad de enseñar a las máquinas a hacer y a pensar, según como nosotros hacemos y pensamos. De esta forma, la interpretación que las máquinas pueden hacer del mundo se estructura de la misma forma que los humanos lo hacen, sujetas a las creencias —morales, socioculturales, políticas, económicas, geográficas— de sus programadores humanos.

Solaas (2019: 15), dice de los artistas programadores que “Establecen restricciones, pero al mismo tiempo abren posibilidades, crean sistemas, pequeños mundos artificiales. Generan, por ende, formas de libertad”. La construcción de algoritmos dentro de la práctica artística da la posibilidad de acercarse al mundo, siguiendo un método que lleva lo práctico y lo teórico a la par, ya que el código computacional es leído por las máquinas a la vez que se produce un efecto visible, lo que abre espacios de reflexión desde la práctica artística con código. En este sentido, se propone al “acto de escritura” como un proceso reflexivo a través del texto que se relaciona directa, aunque no únicamente, con la escritura de algoritmos y la POO, ya que ambos elementos responden a una estructura en la que el código computacional, al ser un texto leído y ejecutado de forma simultánea, provoca un ensamble entre lo que se piensa, lo que se hace y lo que se espera, ensamble que deriva en un juego de traducciones.

Una primera traducción ocurre desde la observación, análisis y posicionamiento ante el mundo del artista —programador, traductor— a la escritura en POO, que es el primer paso para abrir un canal de comunicación entre la máquina y el mundo. En esta traducción se encuentra un proceso de exploración y reflexión por parte del traductor. Después ocurre una nueva, que es la que produce una salida,⁶ que es un testigo de los procesos interpretativos de la máquina y por lo tanto de su programador.

⁶ *Out*, que puede ser la proyección de una imagen, un sonido, un movimiento o una toma de decisiones, sumas, restas, comparaciones, etcétera.

¿QUÉ ES UN “ACTO DE ESCRITURA”? LA POO COMO ESTILO DE ESCRIBIR

Aquí el “acto de escritura” no se problematiza desde el ámbito filosófico. Se trata de construir al “acto de escritura” a manera de un método de investigación desde la creación, desde un “estar haciendo”, en presente continuo. Juan Carlos Arías (2010), propone que la escritura en el arte es un terreno que concibe a la investigación como una práctica creativa y reflexiva, no como una dualidad sino como una resonancia que no se limita a la elaboración teórica de un texto sino al ejercicio de un estilo de escritura que es soporte y contenido del discurso que se desarrolla.

Para Arías (2010: 7):

[...] no existen objetos de investigación en sí mismos, sino que se convierten en objetos de investigación en la medida en que el proceso de investigación misma los configura como tales. Esta definición del objeto de investigación implica necesariamente una distancia [...] Dicha distancia es posible desde la escritura.

El “acto de escritura” provoca un acercamiento desde la distancia a su propia práctica que propone la experimentación con el mundo a través del texto. Así, un “acto de escritura” es conocimiento en su hacer, porque implica una decisión en el estilo, una resonancia intencional que permite construir, desdoblar y conectar conceptos, convirtiéndose en la evidencia de una reflexión que acontece.

La POO, en su estructura reflejo de la realidad autorreferente, y los algoritmos, como estructura de pensamiento interpretativo, conforman un “acto de escritura” puesto que son un estilo de texto con sus propias lógicas y configuraciones. Un estilo pensado desde sus inicios para la experimentación y que por su estructura tiene la potencia de conectar conceptos en su propio acontecer, en el cual se escribe, y se piensa, en un “lenguaje regulativo” (Solaas, 2019), dividido en pasos, cuyo objetivo es abrir canales de comunicación con las máquinas. En este sentido, es notable el trabajo de algunos artistas que construyen discursos cuestionando su presente desde la práctica de la programación. Un ejemplo de esto es el trabajo de la artista británica Anna Ridler, quien ha buscado mantener una relación conceptual fuerte entre el discurso y el medio, como lo podemos ver en una serie de dos de sus trabajos: La pieza de video *Mosaic Virus* (2018) y en la serie fotográfica *Myraid (Tulips)* (2018).

En la pieza *Mosaic Virus*, la artista propone un paralelismo conceptual con la “tulipomanía”, nombre por el que se le conoce a un período acontecido en los años 1630, en el que la especulación del precio de los tulipanes en Países Bajos se disparó de forma desmedida, y la actual especulación en los precios del *bitcoin*.

Como consecuencia de la sobreplantación de tulipanes durante la tulipomanía, se produjo una alteración genética en estas flores que tuvo como consecuencia lo que hoy se

conoce como “el virus del mosaico” —*mosaic virus*—. Esta variación genética rompe con el color sólido natural de los tulipanes, provocando líneas de color que asemejan a flamas sobre sus pétalos, volviéndolos más llamativos para los compradores, lo que contribuyó a mantener alta la especulación en su precio.

Para generar este video, Ridler utiliza una Inteligencia Artificial (IA) de tipo “Red Generativa Antagónica” (GAN, por sus siglas en inglés), un tipo de algoritmos que han tomado mucha relevancia en los últimos años dentro de las propuestas de arte IA. *Mosaic Virus* tiene la particularidad de generar relaciones conceptuales dentro de su construcción reflexiva como pieza.

La IA que genera los tulipanes en la pieza, sujeta las características de las flores al precio especulativo actual del *bitcoin*, actualizando sus ciclos de vida y aumentando o disminuyendo la “cantidad de virus” que las afecta y así volviéndolas “más atractivas” mientras el precio de las criptomonedas cambie. La relación de conceptos que se visibiliza en la imagen, cuestiona los procesos de aprendizaje automático de la IA para conjugar los conceptos de especulación, decadencia y colapso, que son la parte central del discurso de la pieza.



Imagen 6. *Mosaic Virus*. Anna Ridler (2018).

ARTÍCULO

YUNUEN ROSAS ORTIZ

El acto de escritura: La Programación Orientada a Objetos como productora de conocimiento desde la práctica artística

Para entrenar a la IA de *Mosaic Virus* fue necesaria la realización de una serie fotográfica de 10 000 elementos a manera de *set* de entrenamiento, que documentan una variedad amplia de tulipanes según su edad, color y si se encuentra presente o no el virus en ellas. Esta serie fue capturada y clasificada a mano por la artista, un proceso que deja documentado en *Myraid (Tulips)* (2018), la cual tiene la intención de evidenciar la parte humana detrás de la máquina, mostrando sus notas y procesos de clasificación escritos sobre cada fotografía impresa.

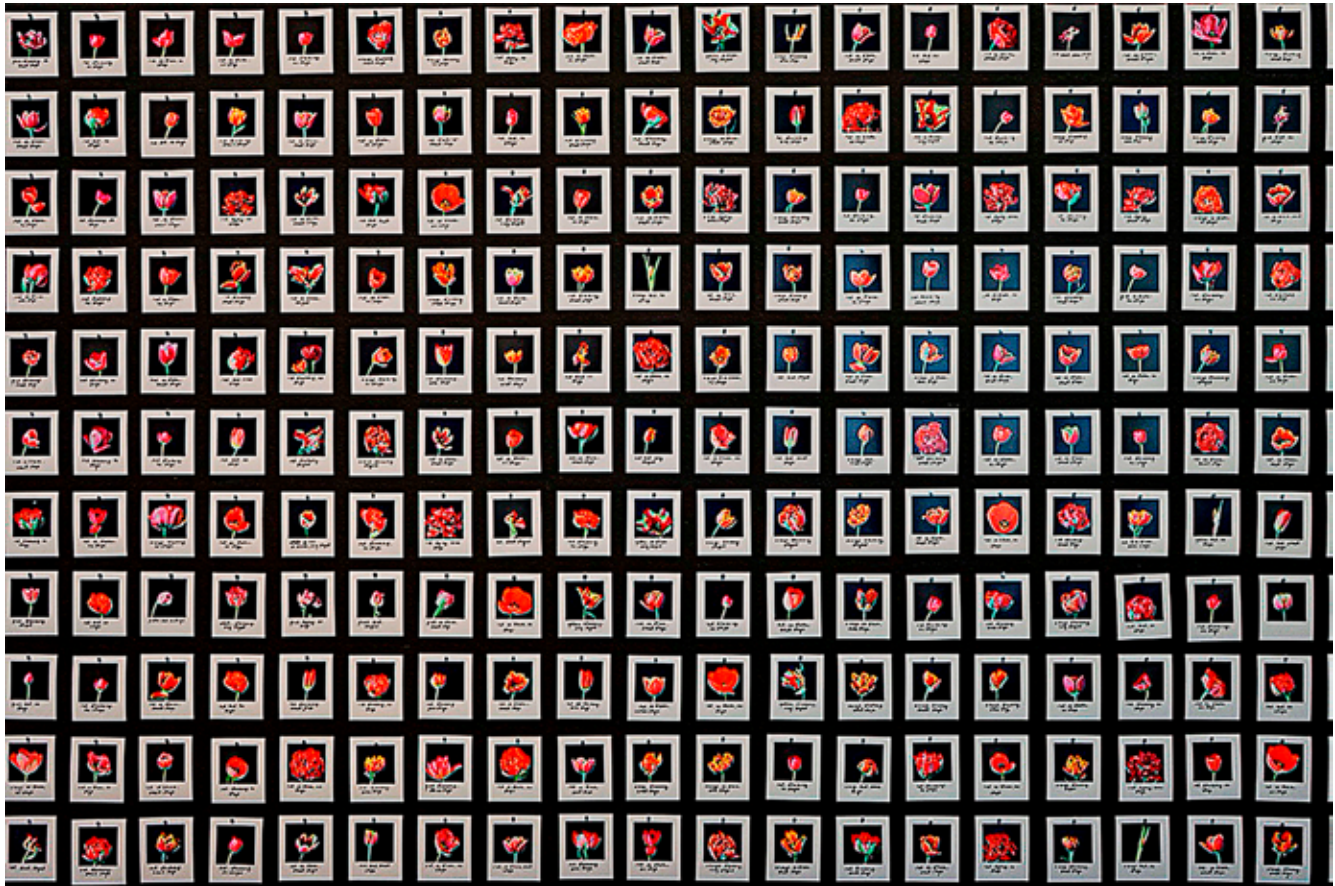


Imagen 7. Vista Myraid (Tulips). Anna Ridler (2018).

En estas piezas se puede observar como un estilo de escritura, en este caso la POO, emiten un eco en el ensamble conceptual de las piezas, a pesar de no permitirnos ver el texto. Anna Ridler declara en una entrevista con Manuel Guerrero: “Realmente quería asegurarme de que estaba usando el aprendizaje automático no solo como una herramienta o proceso, sino como una forma de entender el tema” (Ridler, 2019: 10). A través de los procesos de aprendizaje automático de las máquinas, Ridler abre distancia con el objeto de investigación —la decadencia, la especulación y colapso—, en el caso de *Mosaic Virus*

al permitir un juego entre el valor, la especulación y la manía que se proyectan en el entrenamiento de la máquina y se muestran en la imagen generativa de un tulipán. Y en el caso de Myraid (Tulips) al visibilizar la parte humana que todo algoritmo tiene, por más complejo que éste sea.

Anna Ridler propone, con el diálogo entre ambas piezas, a la POO aplicada en el aprendizaje automático como un método de construcción de mundos e imaginario, desarrollado desde sus propios “actos de escritura” en la práctica artística volviéndose conocimiento al abrir distancia con el objeto de investigación desde su hacer.

[...] tenía sentido usar el aprendizaje automático para hacer esto en lugar de cualquier otro recurso: Podría haber usado animación, video o algo así, pero el uso del aprendizaje automático me permitió aclarar estas cosas conceptuales, entre toda la diferencia de ideas y hacer más clara la investigación que estaba creando (Ridler, 2019: 12).

En prácticas latinoamericanas, también se hace notable en trabajo de Eugenio Tisselli (2017),⁷ un artista mexicano quien afirma que “*Es hora de hacer transparentes esas cajas negras*”,⁸ para esto hace propuestas alrededor de la literatura electrónica a través de poesía algorítmica cuestionando el potencial de las máquinas para escribir usando recursos de internet. También está la artista mexicana Frida Robles⁹ con la pieza titulada *Mi tía abuela* (2018), en la cual se hace un juego entre memorias y código computacional para contar una historia que en conjunto presenta un panorama del trato a mujeres lesbianas en 1970 en México, se hace uso de texto y audio para completar dichos recuerdos. Asimismo, Leonardo Solaas,¹⁰ artista argentino, cuyo trabajo de producción e investigación en torno al arte generativo cuestiona la forma en la que se produce arte e indaga en la posibilidad de que [...] el trabajo con sistemas generativos es una manera de ir investigando cómo sería un cosmos donde el hombre ya no es el centro y el origen de todo, sino apenas un nodo en una compleja red de agentes humanos y no humanos (Solaas, 2018: 43).

¿Cómo la POO puede generar conocimiento desde su práctica? Deleuze y Guattari en “Lo liso y estriado” (2020), proponen modelos de conocimiento desde las figuras de lo “liso” como espacios desconocidos y lo “estriado” como espacios estudiados. En la relación entre ambas partes —liso y estriado— podemos encontrar espacios de conocimiento posibles desde la creación y la exploración. La POO como estilo de escritura provoca “estrías” en el mundo, puesto que su estructura resuena con el ordenamiento de los objetos tangibles; las relaciones que puede establecer entre sus objetos y clases crean espacios

⁷Véase el portafolio del artista en <http://www.motorhueso.net/>

⁸It's time to make those black boxes transparent

⁹Véase el portafolio de la artista en: <https://fridarobles.wordpress.com/>

¹⁰Véase el portafolio del artista en: <http://solaas.com.ar/>

de posibilidad en su propia escritura y lectura, abre entonces “espacios lisos” proponiendo un texto que se potencia en su propio acto de lectura, insinuando un desdoblamiento del mundo. Como “acto de escritura”, la POO se cuestiona a sí misma, a su implicación social, a sus posibilidades teóricas y prácticas, la relación que mantienen las máquinas con el mundo y el mundo con las máquinas, a través de su propio hacer como en el caso del trabajo de Anna Ridler, Eugenio Tisselli, Frida Robles o Leonardo Solaas.

Existen diversos lugares para enfrentarnos a un “acto de escritura” pero la POO puede posicionarse como uno de los más importantes dada nuestra relación con la tecnología y por la resonancia entre el discurso y el hacer que se proponen desde la estructura escrita de la POO.

CONCLUSIONES

Este trabajo propone a la POO como un “acto de escribir” que genera espacios reflexivos dentro de la práctica artística en el que no solo es posible cuestionar los límites de las máquinas y la relación que mantenemos con la tecnología sino que tiene la potencia de ensamblar discursos filosóficos, sociales, culturales y políticos. Trabajos como el de Anna Ridler, Eugenio Tisselli, Frida Robles o Leonardo Solaas, proponen al “acto de escritura” como una ruta de investigación desde el hacer que cuestiona a lo tecnológico, lo sociedad y al arte, presentando una intersección entre lo máquina y lo humano lanzando preguntas como: ¿a qué discursos se acceden desde la tecnología?, ¿qué se puede hacer con una máquina que no puede hacerse de otra manera? o ¿cómo altera lo tecnológico al arte a nivel conceptual, práctico y discursivo?

El arte como campo de conocimiento ha propuesto a la práctica artística como una ruta de acercamiento al mundo. Las posibilidades que ofrece la escritura en la era digital, de algoritmos y programadores se encuentran en constante estado de interrogación por lo que es importante indagar en quiénes y desde dónde se están escribiendo esas posibilidades.* ¶

* Este escrito formó parte del trabajo de titulación de la autora como Licenciada en Arte Digital de la UAEMéx, bajo la dirección del M. en E. V. Leonardo Rodríguez Torres y con la asesoría del M. en E. V. Juan Mojica Arias y la M. en E. V. Sofía Sienna Chaves.

REFERENCIAS

- Canales, L. (2018). "El pequeño arte de traducir". En *Letras Libres*, Issue 240, pp. 73-74.
- Arías, J. (2010). "Editorial. La investigación en artes: el problema de la escritura y el 'método'". En *Cuadernos de música, artes visuales y artes escénicas*, vol. 5, núm. 2, pp. 5-8.
- Ballew, T. y R. F. (2018). *Mi tía abuela*. Disponible en: <http://oral.pub/Mitiaabuela/> [Consultado el 19 noviembre de 2020].
- Deleuze, G. y Guattari, F. (2020). "Lo liso y lo estriado". En: *Mil Mesetas. Capitalismo y esquizofrenia*. Valencia: PRE-TEXTOS, pp. 483-510.
- Grundon, E. (2018). *Vista de Myraid (Tulips) de Anna Ridler*. INFOTEC (n.d.) *JAVA básico*. México: INFOTEC.
- Ridler, A. (2018). *Mosaic Virus*. Disponible en: <http://annaridler.com/mosaic-virus> [Consultado el 5 junio de 2020].
- Ridler, A. (2018). *Myraid (Tulips)*. Disponible en: <http://annaridler.com/myriad-tulips> [Consultado el 6 junio de 2020].
- Ridler, A. (2019). *De lo digital a lo real. Anna Ridler* [Entrevista].
- Robles, F. (2020). *Frida Robles*. Disponible en: <https://fridarobles.wordpress.com/> [Consultado en noviembre 2020].
- Rodríguez, J., Santamaría, L., Rabasa, A. y Martínez, O. (2003). *Introducción a la programación. Teoría y práctica*. Alicante: Club Universitario.
- Solaas, L. (2010). "Generatividad y molde interno". En E. Causa, ed. *Invasión generativa*. La Plata: Invasores de la Generatividad, pp. 9-24.
- Solaas, L. (2017). *Leonardo Solaas*. Disponible en: <http://solaas.com.ar/index.html> [Consultado el 19 noviembre 2020].
- Solaas, L. (2018). *Autómatas creadores: los sistemas generativos en el cruce del arte y la tecnología*. Disponible en: <https://medium.com/@solaas/aut%C3%B3matas-creadores-los-sistemas-generativos-en-el-cruce-del-arte-y-la-tecnolog%C3%ADa-f6d36dc1edd5> [Consultado el 19 noviembre 2020].
- Solaas, L. (2019). *Lógica sensible: observaciones sobre el acto de programar*. Disponible en: <https://medium.com/@solaas/l%C3%B3gica-sensible-observaciones-sobre-el-acto-de-programar-d4c15da0bfa1> [Consultado el 22 marzo 2020].
- Tisselli, E. (2017). *motorhueso.net*. Disponible en: <http://www.motorhueso.net/> [Consultado el 19 noviembre 2020].