

**Propuesta de Proyecto de Investigación**  
**Maestría en Ciencias y Tecnologías de la Información**

<20, noviembre, 2022>

**1. Nombre del proyecto** “Estudios en programación automática como elemento fundamental de la ingeniería de software automatizada”

**2. Responsable(s)**

Humberto Cervantes Maceda, DIE UAM, T-138, hcm@xanum.uam.mx

Eduardo Vázquez Santacruz, DIE UAM, T147, evazquez@xanum.uam.mx

**3. Área(s) de conocimiento relacionada(s) con el proyecto**

Ingeniería de software, aprendizaje de máquinas e inteligencia artificial

**4. Descripción del proyecto**

- Contexto

El área de conocimiento referente a la programación automática tiene como objetivo que el propio software escriba los programas que necesita la sociedad atendiendo los requerimientos e indicaciones de ellas. Esta área del conocimiento se está desarrollando hoy en día en diversos lugares de nuestro planeta con el fin de hacer más cómoda y eficiente la actividad cotidiana de los humanos en general a través de la automatización de la generación de programas computacionales [1,2].

Los programadores han estado trabajando para que su actividad sea lo más sencilla posible y ayude a las personas a crear sus propios programas de una manera amigable simplemente indicando requerimientos a la computadora [2]. En este sentido se han inventado y desarrollado diferentes tipos de estrategias, métodos, técnicas y multitud de programas computacionales con la idea de resolver los diversos tipos de problemas, que se presentaban tanto a la comunidad de programadores como a otras personas, con el apoyo de las computadoras esperando que el trabajo de programar fuera más rentable o simplemente que se pudiera solucionar con mayor facilidad y efectividad en beneficio de la sociedad en general [4,6].

Hoy en día, existen muchos ejemplos de programas que escriben programas. Sin embargo, cuando nos referimos al concepto de programación automática los objetivos son más ambiciosos dado que se pretende que el software genere nuevo software a partir de los requerimientos indicados por los humanos, y eso no es fácil de conseguir [6]. En este trabajo tenemos la visión de contribuir con estudios y análisis para avanzar en estos objetivos.

- Motivación

Se propone realizar estudios y análisis respecto a diferentes estrategias que permitan crear programas computacionales que escriban programas que finalmente generen entidades ejecutables, a partir de las indicaciones humanas. Pretendemos contribuir en el esfuerzo de lograr que las computadoras sean un instrumento de apoyo mediante el cuál se codifiquen programas que por sí mismos generen programas sin la intervención humana y atendiendo únicamente a los requerimientos e indicaciones establecidas por los humanos. Una de las principales dificultades en esta ambiciosa tarea consiste en que actualmente los programas generados por las computadoras son largos y difíciles de comprender. No son tan sencillos como los programas escritos por personas y en general tienen muchas instrucciones repetidas.

En la actualidad se están realizando esfuerzos en conseguir que la computadora ayude en la construcción de programas. Algunas empresas están invirtiendo recursos materiales y humanos para conseguir el logro de estos objetivos. En algunos lugares del planeta existen proyectos de varios años de trabajo y de cooperación conjunta entre el sector académico y empresarial y al parecer los resultados que están obteniendo son muy esperanzadores.

Dado que esta ambiciosa meta está en sus inicios de desarrollo, existe mucho qué hacer hasta conseguir que los humanos podamos transmitir nuestras ideas a las computadoras con el objetivo de que nos ayuden a llevarlas a la práctica a través de la generación de programas computacionales que ejecuten la codificación correspondiente y permitan llevar a cabo actividades en concreto. En este trabajo se propone realizar estudios y análisis pertinentes para contribuir en esta importante área del conocimiento.

- Aporte esperado al área de conocimiento

- Estudios complementarios para diseñar y construir programas computacionales que codifiquen a su vez otros programas computacionales que finalmente generen entidades ejecutables
- Análisis complementarios para diseñar y construir programas computacionales que codifiquen a su vez otros programas computacionales que finalmente generen entidades ejecutables
- Nuevas estrategias para diseñar y construir programas computacionales que codifiquen a su vez otros programas computacionales que finalmente generen entidades ejecutables

## 5. Objetivos

- Objetivo general

Estudiar diversas técnicas y estrategias para diseñar y construir programas computacionales que codifiquen a su vez otros programas computacionales que finalmente generen entidades ejecutables

- Objetivos particulares

- Estudiar el Estado del Arte
- Diseñar e implementar estrategias basadas en aprendizaje automático e inteligencia artificial en general para diseñar y construir programas computacionales que codifiquen a su vez otros programas computacionales que finalmente generen entidades ejecutables
- Evaluar y comunicar los resultados

## **6. Metodología**

El proyecto se desarrollará considerando las siguientes etapas,

- Planteamiento del problema
- Especificación de requerimientos
- Diseño de la propuesta de solución
- Implementación de la propuesta de solución
- Pruebas y ajustes de la propuesta de solución
- Desarrollo de la documentación a lo largo del desarrollo del proyecto
- Desarrollo de la idónea comunicación de resultados (ICR)
- Revisión y ajustes de la ICR
- Preparación y presentación del examen de grado

## **7. Calendarización de actividades**

El proyecto se desarrollará considerando el siguiente programa de actividades,

Primer trimestre.

- Planteamiento del problema
- Revisión del Estado del Arte
- Documentación

Segundo trimestre.

- Diseño de la propuesta de solución
- Desarrollo de la solución
- Documentación

Tercer trimestre.

- Desarrollo de la solución
- Primera versión de la idónea comunicación de resultados (ICR)

Cuarto Trimestre.

- Revisión de la idónea comunicación de resultados (ICR)
- Ajustes al ICR
- Preparación y presentación del examen de grado

## **8. Infraestructura necesaria y disponible**

Una computadora personal

## **9. Lugar de realización**

Modalidad mixta (presencial y/o a distancia)

## 10. Entregables

- Memoria en extenso en congreso nacional o internacional
- ICR.

## 11. Referencias bibliográficas básicas

- [1] Andrew Berlin and Daniel Weise, "Compiling Scientific Code Using Partial Evaluation", IEEE Computer, vol. 23, no. 12 (Dec. 1990), pp. 25-37.
- [2] Charles Rich and Richard C. Waters, "Automatic Programming: Myths and Prospects", IEEE Computer, vol. 21, no. 8 (Aug. 1988), pp. 40-51.
- [3] Gregor Kiczales et al., "Aspect-Oriented Programming," Proc. Euro. Conf. OOP 1997, LNCS 1241, Springer Verlag, 1997.
- [4] Herbert A. Simon, "Whether Software Engineering Needs to be Artificially Intelligent", IEEE Trans. on Software Engineering, vol. SE-12, no. 7 (July 1986), pp. 726-732.
- [5] K. Martineau, "*Toward artificial intelligence that learns to write code*," MIT News, June, 2019.
- [6] Mitchell, T.M. (1997). Machine Learning. McGraw Hill
- [7] Ricardo Aler Mur, "Automatic Inductive Programming", ICML 2006 Tutorial. June 2006
- [8] Robert Paige and Shaye Koenig, "Finite Differencing of Computable Expressions", ACM Trans. on Programming Languages and Systems, vol. 4, no. 3 (July 1982), pp. 402-454.