
El pensamiento analítico - asociativo en la enseñanza y aprendizaje de la Programación.

Analytical - associative thinking in teaching and learning programming

Juan Carlos Fonden Calzadilla

Facultad de Ciencias Médicas “Dr. Miguel Enríquez”. La Habana. Cuba

ORCID: <https://orcid.org/0000-0001-7478-8628>

Correo(s) electrónico(s):

fonden1980@gmail.com

Recibido: 15/08/2020

Aceptado: 12/02/2021

Resumen: Recurriendo a los métodos de investigación análisis documental, modelación, enfoque de sistema y la observación, se realizó un estudio del proceso de enseñanza y aprendizaje de la Programación en las universidades José Eduardo Dos Santos” de Angola y la Universidad Tecnológica de la Habana “José Antonio Echeverría”; constatándose insuficiencias al relacionar clases, objetos y sus códigos asociados y como consecuencia bajos rendimientos académicos. Como resultado se avala la necesidad de elaborar recomendaciones metodológicas aplicables a ambos contextos educativos que contribuyan a la formación de un pensamiento analítico y asociativo que posibilite la adquisición de resultados docentes superiores en esta disciplina.

Palabras clave: Pensamiento analítico; Pensamiento asociativo; Programación; Recomendaciones metodológicas.

Abstract: Using research methods, documentary analysis, modeling, system approach and observation, a study of the teaching and learning process of programming was carried out at the José Eduardo Dos Santos ”universities in Angola and the “ José Antonio Echeverría ”; finding insufficiencies when relating classes, objects and their associated codes and as a consequence low academic performance. As a result the need to develop methodological recommendations guarantees applicable to both educational contexts that contribute to the formation of an analytical and associative thinking that enables the acquisition of higher learning outcomes in this discipline.

Keywords: Analytical thought; Associative thought; Programming; Recommendations metodológicas.

Introducción

El aprendizaje de la Programación, elemento fundamental en la formación informática del profesional, es un proceso continuo, complejo y planificado. Su éxito obedece a que se planeen, organicen, ejecuten y controlen sistemas de tareas docentes con una adecuada relación teórico-práctica, en el cual se integren elementos motivacionales, afectivos, cognitivos y valorativos, en conformidad con las exigencias y necesidades del plan de estudio y los objetivos de la asignatura. Su estudio es considerado una actividad difícil debido a la complejidad implicada en su desarrollo.

Existen educandos que no logran adquirir las habilidades necesarias para la programación, incluso después de terminar un curso introductorio de algoritmización, “un estudio estima que entre un 25 a 80 por ciento de los estudiantes en Estados Unidos abandonan sus primeras clases de programación debido a la dificultad que enfrentan para aprender a programar” (Insuasti, 2016, p.237) y los profesores de esta disciplina informática, perennemente expresan la necesidad de hacer algo para contrarrestar estos bajos resultados académicos, cuestión en permanente debate.

Insuasti (2016) y otros autores consideran que el desarrollo preliminar de algunas habilidades del pensamiento como el análisis, la síntesis y la abstracción facilitaría el aprendizaje de la Programación.

Por otra parte, el autor ha constatado, desde el curso 2009 – 2010, hasta la actualidad, que en las Universidades Tecnológicas y otros institutos educacionales, en el Proceso de Enseñanza y Aprendizaje (P.E.A) de la Programación, persisten bajos resultados académicos y entre sus causas se observa la dificultad para *analizar* e identificar con exactitud los *vínculos y asociaciones* entre clases o entidades del mundo real y sus códigos asociados; escribir instrucciones directamente en el ordenador sin un previo análisis y las fallas al establecer relaciones entre componentes y subcomponentes en la realización de un software.

Las situaciones problemáticas antes mencionadas sugieren la *necesidad de continuar investigando para obtener resultados académicos superiores en el PEA de la Programación*.

Para el desarrollo de este artículo se proyectaron y cumplieron las siguientes tareas:

Sistematización de los estudios realizados sobre el PEA de la P.O.O en la red de redes y otras fuentes bibliográficas; Diagnóstico del Proceso de Enseñanza – Aprendizaje en las asignaturas Programación Estructurada, Programación .Orientada a Objetos (P.O.O) y Estructuras de Datos en las Facultades de Ingeniería Informática e Industrial en la Universidad “José Eduardo Dos Santos” de la República de Angola y la Universidad Tecnológica de la Habana “José Antonio Echeverría”; Identificación de las particularidades de la enseñanza de la P.O.O y entrevistas a programadores, profesores y estudiantes; Análisis de los resultados docentes en los exámenes de P.O.O, indicaciones metodológicas, libros, folletos con ejercicios resueltos y guías para los estudiantes.

Por todo lo expuesto, se continúa indagando para evidenciar si la formación de un pensamiento analítico y asociativo contribuye a solucionar las insuficiencias detectadas, si potencia la solución de problemas complejos y la obtención de resultados académicos superiores. A continuación se analizarán diferentes vínculos entre entidades y se elaboraran orientaciones metodológicas a fin de aplicarlas al PEA de la Programación.

Desarrollo

La enseñanza de Programación en las escuelas no es una actividad nueva. Lenguajes de programación como Fortran, Lisp Cobol Y Logo surgieron a finales de los años 1955 y se convirtieron en poderosas herramientas para los jóvenes programadores que iniciaron esta tarea. El PEA de la Programación tiene, entre sus objetivos esenciales, la formación y desarrollo de habilidades, que posibilite la resolución de problemas del ámbito escolar, profesional o de la vida práctica (Cortés, 2017).

De igual forma, se reconoce ampliamente el aporte que proporciona el aprendizaje de la Programación a la formación del pensamiento lógico y abstracto en los educandos, que desde edades tempranas son entrenados en estas tareas (Cortés, 2017).

La (P.O.O), paradigma de programación basado en objetos y sus interacciones, que simula la forma en que el ser humano vive e interactúa con las cosas y sucesos del mundo real, donde se identifican, modelan y programan acciones sobre clases relacionadas y/o asociadas entre sí. Es un paradigma popularizado a partir de 1990 e introdujo una nueva forma de constituir el código de un programa informático. Actualmente existe una gran variedad de lenguajes de programación que emplean el paradigma orientado a objetos.

Otras razones para impulsar el estudio de la Programación en edades tempranas es el desarrollo del pensamiento sistémico, la identificación de clases, atributos y variables en un problema, el trabajo colaborativo, la autogestión del conocimiento, el aprendizaje a partir del error, la modelación de disímiles estrategias para la solución de problemas, la toma de decisiones fundamentadas en razonamientos lógicos y la elaboración de proyectos informáticos.

Por las ideas expresadas, hoy día, en países como Francia, Estonia, Reino Unido, Finlandia, Australia y Alemania se estimula el aprendizaje de la Programación. Desde hace varios años, diversas naciones han comenzado a adoptar a la programación como

parte del currículo común de los planes de estudio, al comprenderse que esta disciplina es necesaria para el desarrollo intelectual de los estudiantes.

En el presente trabajo, un paradigma de programación constituye un enfoque específico u ontología para diseñar soluciones (Insuasti, 2016). Los paradigmas difieren unos de otros en los conceptos y la forma de abstraer los elementos implicados en un problema, así como en los pasos que componen su solución. También queda establecido que el término relacionar es sinónimo de *asociar, vincular, combinar, conectar, enlazar y acoplar*.

Además, una entidad es un elemento, proceso o suceso reconocido, con información identificable, ejemplos: una persona, empresa, silla, cuenta bancaria, un planeta, etc., del mundo real, y como tal posee ciertos atributos que la identifican y diferencian de las demás entidades (Fonden, 2019). Las entidades son objeto de estudio en la disciplina Programación Orientada a Objetos.

Una clase es algo similar a las entidades. Una clase, es un modelo utilizado para crear nuevos objetos y no existe de manera aislada. Todo objeto o entidad es una instancia o ejemplar de una clase. La posibilidad de crear jerarquías entre las clases es una particularidad que diferencia la P.O.O de otros paradigmas de programación, debido a que posibilita extender y reutilizar el código existente (fig. 3).

Una clase se representa en el diagrama de clases UML como un rectángulo dividido en tres partes, (fig. 1):

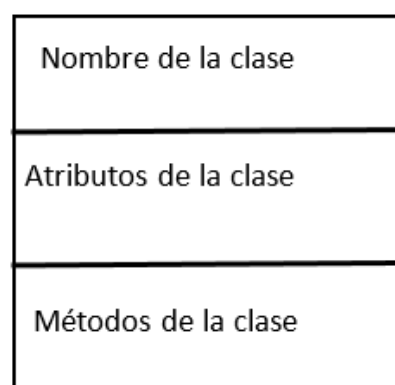


Figura No1. Representación gráfica de una clase o una entidad. Elaboración propia.

El pensamiento analítico y asociativo. Ejemplos.

Analizar es un término usado constantemente en diferentes contextos de la vida. El análisis es una actividad intelectual que requiere de la descomposición de un objeto o

proceso en todas sus partes integrantes, así, el análisis del Proceso de Enseñanza y Aprendizaje de una asignatura implica la observación del contenido, los métodos, las formas de evaluación, las formas organizativas, el profesor y los estudiantes, entre otros elementos.

El análisis se opone lógicamente a la síntesis por ser una operación del pensamiento que consiste en la composición ordenada de los elementos diferentes de un todo, operación contraria a la analítica, no obstante ambas se complementan.

En ocasiones, en los procesos de análisis, existen obstáculos derivados de modelos mentales o estereotipos predefinidos que impiden alcanzar soluciones, estos modelos se alejan del verdadero análisis, por tal razón, “la raíz de los problemas en el pensamiento con frecuencia está en las suposiciones falsas. Ya que las suposiciones por lo general son inconscientes, a menudo incorporan prejuicios, estereotipos y falsas creencias arbitrarias” (Elder y Paul 2003, p.14).

Al mismo tiempo, en la docencia, como en la naturaleza, la sociedad y el pensamiento, se vinculan los saberes, materias, asignaturas, disciplinas, ciencias y esos vínculos originan nuevas materias y ciencias (Lazo, Valcárcel y González 2015; Pedroza, 2006). Así, la informática es el tratamiento automático de la información con el empleo de computadoras; la bioquímica es una ciencia interdisciplinar, que extrae sus temas de interés de muchas disciplinas y así sucede con otras ciencias como la economía política y las telecomunicaciones (Rosental y Ludin, 1984).

Estos vínculos se fundamentan, filosóficamente, en una ley general conocida como Ley de la Concatenación Universal de todos los objetos y fenómenos y otras leyes de la dialéctica. “Las conexiones entre los objetos y fenómenos pueden ser directas e indirectas, constantes y temporales, esenciales y no esenciales, casuales y necesarias, funcionales” (Rosental y Ludin 1984, p.76).

A continuación, el autor analiza un conjunto de posibles vínculos del objeto o entidad Estudiante, en interacción con otras entidades:

1. Un Estudiante se relaciona consigo mismo, puede ser jefe de grupo o subordinado. Ambos, el que dirige y el subordinado es un estudiante. A este tipo de relación se le llama clases involutivas en P.O.O. También es conocida como relación reflexiva. En matemática, una relación reflexiva es aquella donde cada elemento se relaciona

- consigo mismo. El estudiante se relaciona consigo mismo en todo momento de la vida;
2. El Estudiante se relaciona temporalmente (transitoriamente) con un Profesor. Relación del tipo 1:1. De uno a uno, determinada por un tiempo. Esto puede acontecer durante una clase, una consulta o un examen. También puede relacionarse establemente (durante un semestre, un curso escolar o más) con uno o varios profesores. Relación del tipo 1: M. De uno a muchos. Esta relación es muy conocida y frecuente, pues el estudiante también se relaciona con muchos estudiantes y muchos otros objetos de la vida;
 3. El Estudiante *es un* becado (interno) o externo. La relación, *es un*, es relación *de herencia* en Programación y constituye una especialización – generalización. Es el mecanismo por el cual una clase hereda las características y métodos de otra clase;
 4. Un Estudiante, en el laboratorio, en una hora determinada está realizando un examen. Aquí intervienen las entidades, Estudiante, Laboratorio y Hora, que al relacionarse dan origen a una nueva entidad llamada Examen y se conoce como agregación;
 5. Muchos Estudiantes se relacionan con muchos Profesores. Relación del tipo M: M. De muchos a muchos, muy empleadas en bases de datos relacionales;
 6. El estudiante se relaciona con cosas o entidades del mundo real, entre ellos, su computadora, el dinero, los alimentos, el carro y el clima, se relaciona con sus pensamientos y emociones. Relaciones y asociaciones infinitas;
 7. ¿Qué universo de asociaciones resultará al interactuar simultáneamente las entidades Profesores; Disciplinas o asignaturas; Departamentos; Facultades; Aulas; Laboratorios; Exámenes; Padres y otras? (fig. 2 y 3). Cada momento en que dos clases se unen para trabajar juntas y alcanzar un fin, se produce una asociación, que dará lugar a otras.

En este artículo se asume pensamiento asociativo como la *capacidad intelectual que tiene el ser humano de constituir y relacionar ideas, opiniones, conceptos, razonamientos y representaciones en su mente*, (Fernández, 2016) caracterizándose, en el proceso de enseñanza y aprendizaje, por presentar el contenido entrelazado (combinado, asociado, vinculado, conectado), como un todo, en función de alcanzar el sistema de conocimientos, habilidades y valores a cumplirse en el plan de estudios de la carrera.

El pensamiento asociativo permite la solución de problemas complejos y el diseño de nuevos productos y conocimientos valorizados, pues lo concibe todo conectado, lo que a la vez es lo más difícil de aceptar, porque, superficialmente el mundo puede parecer desconectado, caótico e inconexo.

Las asociaciones pueden ser *temporales* o *permanentes*, en dependencia del contexto, tal como acontece en la vida. Una persona se relaciona temporalmente con la impresora, con el doctor o el abogado y permanentemente consigo misma, con la persona que comparte la habitación y con su mascota.

Las asociaciones mentales son más estables cuanto más significado tienen para el sujeto. Así, recordar algo depende, en gran medida, del interés que se tiene para hacerlo y de las vivencias adquiridas de otros sucesos similares.

El pensamiento asociativo analiza los sucesos desde el punto de vista de las múltiples interacciones, internas o externas, permanentes o transitorias.

Contrario al pensamiento asociativo, es el secuencial, en el que se percibe un encadenamiento lineal de los sucesos y factores. Si bien una amplia gama de problemas pueden ser resueltos desde la perspectiva de un pensamiento secuencial, otros necesitarán del establecimiento de múltiples asociaciones entre objetos, procesos y fenómenos para su comprensión, tanto en la vida cotidiana como en el aprendizaje de la Programación.

Además de todo lo expresado hasta aquí, un elemento a tener en cuenta ante un problema es el conocimiento significativo de la entidad identificada, es decir, la experiencia previa adquirida por el estudiante sobre el proceso en el que se encuentra la entidad. Si la entidad es la Cuenta Bancaria de un Cliente en un Banco, se necesita un conocimiento previo o la búsqueda de información del funcionamiento de un Banco y su interacción con las entidades, Cuenta Bancaria y Cliente.

El desconocimiento absoluto del proceso donde se encuentran las entidades identificadas acarrea errores al relacionarlas. El tema en estudio debe ser significativo para el educando, es decir, potencialmente relacionable con la estructura cognitiva del que aprende, y de esta forma puede crear asociaciones mentales.

Asociación entre emociones y resultados de aprendizaje. Opiniones.

Las emociones influyen en el aprendizaje, por lo que se aconseja que en este proceso se promueva el disfrute, la autoestima y se eviten las emociones negativas como el miedo.

La asociación duradera detectada entre las emociones y los resultados de aprendizaje debe ser más estudiada y tomada en cuenta.

Se aplicaron instrumentos de indagación a 236 estudiantes distribuidos en 8 grupos en la Universidad “José Eduardo dos Santos” de la provincia de Bie en Angola durante los cursos escolares 2016, 2017 y 2018 y a 124 estudiantes distribuidos en 4 grupos de segundo año de la carrera Ingeniería Industrial de la Universidad Tecnológica de la Habana, en diferentes momentos.

Se le preguntó a los 124 estudiantes de Ingeniería Industrial, al concluir un examen parcial lo siguiente:

1. Expresa su opinión sobre la asignatura P.O.O
2. Diga si usted se siente complacido con sus profesores de conferencia y clases prácticas.
3. ¿Considera que esta asignatura contribuye a tu formación profesional?

Más del 60% de los estudiantes respondieron positivamente a las tres preguntas y he aquí una de las opiniones expresada por ellos:

Me gusta la asignatura, las clases conferencias las comprendo bien, aunque disfruto más las clases prácticas porque el profesor tiene mucha paciencia, buen trato y nos dice cómo aplicar el conocimiento.

Por otra parte, un estudiante angolano refiriéndose a su profesor de una disciplina informática escribió en una red social *nuevamente no me canso de reconocer y agradecer por la grande persona que es, siempre listo para enseñar y aprender, detallista, prudente, gran compañero y amigo. Estaré siempre agradecido.*

Otro profesor que trabajó con los antedichos alumnos en Angola fue congratulado por sus estudiantes con frases como *el profesor nos ayudó a pensar y siempre lo tendremos presentes porque hemos aprendido más de lo que pensábamos y otros decían “ya podemos crear nuestra empresa de software”.*

En la enseñanza de las ciencias la información sobre el papel de las emociones en el aprendizaje, es limitado, en relación con esto, el autor recomienda reflexionar respecto a lo expresado por Einstein en el periódico The New York Time el 5 de octubre de 1952:

No basta con enseñar a un hombre una especialidad. Aunque esto pueda convertirle en una especie de máquina útil, no tendrá una personalidad

armoniosamente desarrollada. Es esencial que el estudiante adquiriera una comprensión de los valores y una profunda afinidad hacia ellos. Debe adquirir un vigoroso sentimiento de lo bello y de lo moralmente bueno. De otro modo, con la especialización de sus conocimientos más parecerá un perro bien adiestrado que una persona armoniosamente desarrollada. (Memorias Convención Internacional de Salud Pública, 2012, p.1)

Otras indagaciones realizadas

El autor, como partícipe y observador del PEA de la Programación, elaboró una guía de observación para constatar la concreción de acciones encaminadas a la formación de un pensamiento analítico y asociativo en la docencia. En ella se tomó información de las siguientes interrogantes:

1. ¿Es un estilo de trabajo el ejemplificar el contenido de la P.O.O a partir de fenómenos abordados en otras materias o disciplinas?
2. ¿Se vincula el contenido de la Programación con vivencias de la vida cotidiana de los estudiantes?
3. ¿Proporciona un tipo de aprendizaje que lleva al alumno a realizar conjeturas, suposiciones, preguntas o hipótesis sobre la materia recibida en Programación, vinculándola con conocimientos de otras disciplinas y de la vida?
4. ¿Establece relaciones entre los aspectos estudiados en una clase con otros conocidos por ellos en grados anteriores?
5. ¿Tiene en cuenta la relación que poseen los hechos y fenómenos que imparte con leyes o teorías que por su nivel de generalización necesitan de asociaciones entre materias, asignaturas o disciplinas?

En las 30 visitas a clases realizadas a profesores en la asignatura P.O.O. y otras materias durante el curso escolar 2018 - 2019, en la facultad de Ingeniería Industrial, el 90% incumplió parcial o totalmente las interrogantes controladas. Casualmente, los resultados de evaluación de los estudiantes de un examen parcial, que recibían clases de los profesores controlados en el curso escolar 2018 – 2019, arrojaron los siguientes resultados:

- a) Más del 90% de los 130 evaluados identifican todas las clases implicadas en cada temario;

b) Entre el 40 y el 50% de los evaluados fracasó en la identificación certera de las asociaciones entre las clases y los códigos de programación que de ellas se emanan (fig. 3).

Algunos resultados investigativos sugieren que la asociación de ideas y/o sucesos se adquiere en edades tempranas y se ejercita a lo largo de la vida, en este sentido, el análisis de los datos de esta investigación constata que estudiantes mayores de 40 años, de los cursos para trabajadores, fueron más proclives a afrontar problemas en la P.O.O, especialmente al relacionar clases, objetos y códigos de programación asociados.

En la Universidad “José Eduardo Dos Santos”, se hicieron sucesivas comparaciones entre los estudiantes del curso regular diurno y el curso para trabajadores, en la especialidad de Informática, al identificar y asociar clases para construir el código de programación que resuelve el problema. En ambos cursos se obtuvieron bajos resultados académicos, aunque muy bajos en los estudiantes del curso para trabajadores, inferiores al 30%. En el caso de estudiantes adultos y trabajadores con edad mayor a 45 años su rendimiento académico fue inferior al 20%.

Relaciones entre clases. Modelación gráfica de la asociación y herencia.

En P.O.O, dos objetos o entidades pueden interactuar entre sí, relacionarse de diferentes formas, entre ellas: La relación de asociación, tipo de relación básico entre dos clases, cuando un objeto accede a los atributos y métodos de otro; las relaciones de herencia o generalización y las relaciones de dependencia que indican la relación existente entre dos clases, donde una depende de otra para realizar su trabajo, en una posible relación temporal. También existe un tipo de relación fuerte en el cual una clase contiene otra y la contenida carece de sentido sin la clase contenedora, a este tipo de relación se les llama clases anidadas. A continuación, las representaciones gráficas de las relaciones de asociación y herencia (fig. 2, 3).

Existe un tipo de relación fuerte llamada composición, donde el tiempo de vida de un objeto determina el tiempo de vida de otro contenido. Ejemplo: Un libro y sus hojas. Si se destruye el libro, dejan de existir sus hojas.

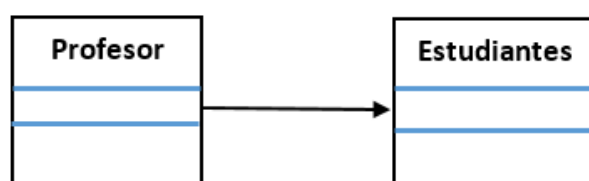


Figura No2. Relación de asociación entre las clases Profesor y Estudiantes. Elaboración propia.

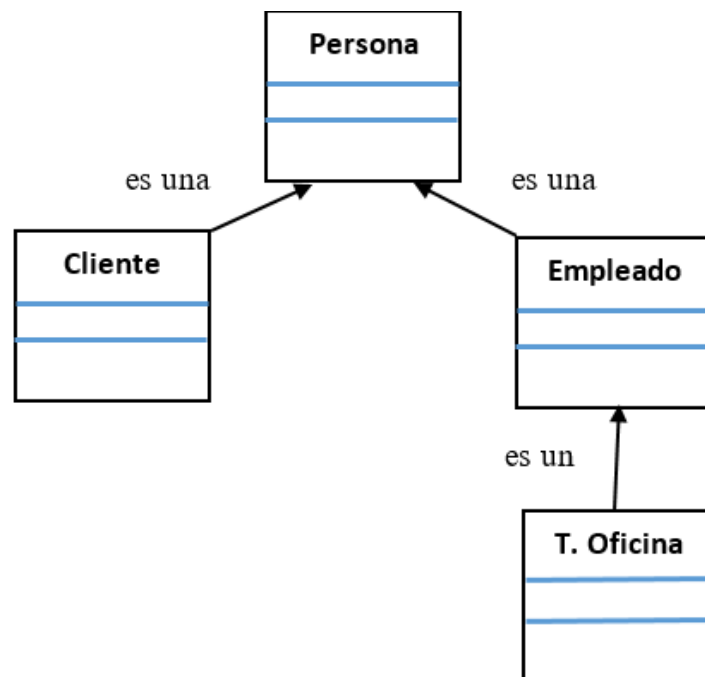


Figura No3. Relación de herencia entre las clases Persona, Cliente y Empleado. Elaboración propia.

Recomendaciones metodológicas.

La sistematización realizada por el autor a los estudios de (Rosental y Ludin, 1984; Elder y Paul, 2003; Fernández, (2016), Lazo, Valcárcel y González, (2015) y Acevedo, (2004)), los resultados obtenidos de la guía de observación a clases, las encuestas realizadas y los exámenes parciales, unido a su experiencia profesional como investigador y profesor de Programación le permitió concluir que el pensamiento analítico y asociativo es necesario para obtener resultados académicos superiores en el PEA de la Programación, aunque no surge de la espontaneidad.

Por consiguiente, sea que tengas un problema textual o expresado en un diagrama y debas identificar partes del todo, relaciones entre partes, internas o externas. Así pues, si tienes que relacionar entidades a fragmentos de códigos de programación o a las partes de un software. Las siguientes recomendaciones metodológicas, encaminadas a profesores y estudiantes, te ayudaran a conseguirlo:

- Propicie, ante todo, una atmosfera de colaboración, respeto, alegría y satisfacción al abordar cada contenido de Programación en el aula o local de estudio. Evite las críticas innecesaria y favorezca la autoconfianza;
- Lea, examine e interprete el problema docente o el proceso a codificar, identifique sus partes principales y secundarias, y sus relaciones. Recopile la información relevante y descubra sus propiedades. Analice el problema desde diferentes puntos de vistas y enlace los elementos que aparentemente no se relacionan. Piense sin restricciones;
- Analiza y reflexiona integralmente para interpretar de forma armónica el problema docente. Identifica los elementos compuestos por el todo, examina sus propiedades e interrelaciones;
- Sintetiza, luego de unir las partes integrantes del todo, expón lo esencial con claridad y precisión. La síntesis es parte del análisis.
- Verifique la existencia de errores o incoherencias en cada parte identificada en el problema o proceso. Realiza deducciones lógicas de los datos;
- Identifique las relaciones permanentes y temporales entre las partes, es decir, entre clases, atributos, operaciones o partes de un programa y lo común y diferente entre los componentes de uno o varios softwares;
- Elabore mapas conceptuales para relacionar conceptos e ideas;
- Modele teórica o gráficamente los elementos identificados y la conformación del todo, para estructurar y promover el proceso de aprendizaje del estudiante. Esta idea es apoyada por Arias (2016), entre muchos autores. La modelación es clave para introducir nuevos conceptos y desarrollarlos. (fig 3);
- Modele teórica o gráficamente cada solución a programar, mediante algoritmos y/o diagramas. Un buen algoritmo es tan necesario y esencial como un buen código de programación (Manovich, 2017). Establezca la transición progresiva del algoritmo al código de programación;
- Relaciónate con profesores y estudiantes de otras universidades, con diferentes puntos de vistas al recibido en clases y dialoga con ellos sobre tu proyecto;
- Disfruta el trabajar con otros, en equipo, en el que exista diversidad de criterios y la colaboración conscientemente organizada entre personas, tecnologías y disciplinas

que cooperan para obtener un fin común. Aplica métodos de enseñanza y aprendizaje que emplean profesores de otras asignaturas o disciplinas, a lo que se le llama cooperación desde el método;

- Evite, al trabajar con otros, las inferencias erróneas “por la cual uno concluye que algo es cierto basándose en qué otra cosa sea cierta, o aparenta ser cierta” (Elder y Paul 2003, p. 47). Las inferencias emergen de las suposiciones y si las suposiciones son incorrectas, también lo serán las inferencias. Verifica y cuestiona las suposiciones. El hecho es la realidad y la suposición es lo que se cree que es el hecho, sin reunir todas las pruebas necesarias. Antes, analiza y verifica;
- Examina frecuentemente la secuencia y coherencia entre los contenidos, es decir, las relaciones intermaterias e interasignaturas;
- El proceso de análisis y síntesis, unido a la identificación de las relaciones intermateria, interasignaturas e interdisciplinarias, propicia el pensamiento analítico – asociativo.

Aplicación de las recomendaciones metodológicas

Se ha realizado, en la carrera de Ingeniería Industrial, una introducción parcial de las recomendaciones metodológicas antes identificadas en la práctica educativa, durante el curso 2018 – 2019, en la asignatura Diseño de Soluciones Informáticas del nuevo plan E. En este sentido, los análisis docentes de los resultados, mostraron una tendencia a la recuperación, tanto desde lo cuantitativo como en lo cualitativo, por la superior calidad de los proyectos informáticos presentados y sus defensas en los tribunales. Se pudo evitar en un alto grado, que estudiantes de otras especialidades o carreras realizaran los códigos de programación y el diseño de la base de datos en los proyectos.

Se mejoró también en la calidad de los informes digitales, los resultados cuantitativos de la disciplina, en lo que concierne a las notas finales y disminución de los reprobados.

En reuniones metodológicas realizadas en el curso 2018 – 2019 con profesores de la asignatura, se debatieron los elementos esenciales expresados en las recomendaciones metodológicas, constatándose un alto nivel de aprobación por los docentes, que se refleja en las preparaciones de clases y el análisis y diseño de los proyectos informáticos.

Conclusiones

La introducción parcial de las recomendaciones metodológicas en la práctica educativa en la carrera de Ingeniería Industrial, durante el curso 2018 – 2019, mostró una tendencia a la recuperación, en cuanto a la adquisición de un pensamiento analítico y asociativo en el PEA de la Programación, tanto desde lo cuantitativo como en lo cualitativo.

La Programación Orientada a Objetos (P.O.O) pone de relieve un cúmulo de vínculos y asociaciones entre entidades. Para transitar con éxito por este proceso se necesita de la formación de un pensamiento analítico – asociativo, y la colaboración organizada entre personas, tecnologías y disciplinas que cooperan para obtener un fin común.

Para un buen desarrollo del PEA de la Programación, se precisa haber adquirido un determinado nivel de análisis y recíprocamente este se alcanza y desarrolla transitando por dicho proceso.

El análisis que se realiza acerca de los posibles vínculos que se dan entre una, dos o más entidades y las recomendaciones metodológicas expresadas, potencian el desarrollo de un pensamiento analítico, asociativo y analítico – asociativo y como resultado final, resultados académicos superiores en el PEA de la Programación.

La P.O.O es una disciplina que prepara a los educandos para el trabajo en equipos, enfrentar retos, resolver problemas complejos, gestionar conflictos, elaborar proyectos, desarrollar la perseverancia y asociar ideas, proceso y fenómenos de la vida.

Referencias bibliográficas

- Acevedo, J. A. (2004). El papel de las analogías en la creatividad de los científicos: la teoría del campo electromagnético de maxwell como caso paradigmático de la historia de las ciencias. *Eureka sobre Enseñanza y Divulgación de las Ciencias*, 1(3). <https://revistas.uca.es/index.php/eureka/article/view/3947>.
- Arias, L. A. (2016). Lenguaje de modelamiento unificado (UML) para modelamiento de embotelladora. *Scientia et Technica*, 21(1). <https://www.redalyc.org/pdf/849/84950584006.pdf>.
- Cortés, J. (2017). Educación. La importancia de que los niños aprendan a programar. *El país*. 6 de Abril de 2017. <https://retina.elpais.com>.
- Elder, L. y Paul, R. (2003). Pensamiento analítico *The Foundation for Critical Thinking*. <https://www.criticalthinking.org>.

- Fernández, C. (2016). Pensamiento relacional en primaria: el papel del maestro. *Uno. Revista didáctica de las matemáticas*, (73).
https://rua.ua.es/dspace/bitstream/10045/65269/1/2016_Fernandez_Ivars_UNO.pdf.
- Fonden, J.C. (2019). Importancia del pensamiento abstracto. Su formación en el aprendizaje de la Programación. *EduSol*, 20(72).
<https://edusol.cug.co.cu/index.php/EduSol/article/view/1195/2499>.
- Insuasti, J. (2016). Problemas de enseñanza y aprendizaje de los fundamentos de programación. *Educación y desarrollo social*, 10(2).
<https://revistas.unimilitar.edu.co/index.php/reds/article/view/1966>. DOI: org/10/18359/reds.1701.
- Lazo, M.A., Valcárcel, N. y González, T.R. (2015). Modelo de Superación con enfoque interdisciplinario en tecnologías de la Salud. *Revista cubana de tecnología y salud*, 6(4). <https://www.medigraphic.com>
- Manovich, L. (2017). Los algoritmos de nuestras vidas. CIC. *Cuadernos de Información y comunicación*, 22. <https://www.redalyc.org/>.
- Rosental, M. & Ludin, P. (1973). *Diccionario filosófico. Representación*. Editorial Política.