

# Competencias específicas en los estudiantes de programación del Instituto Universitario Politécnico Santiago Mariño

Alfredo J. Díaz-Pérez\*

Luisa A. Serra-López\*\*

---

## Resumen

En virtud de los cambios constantes en el entorno económico, político y social que vive el país, los profesionales deben esforzarse por ser más competitivos y, para ello, es necesario lograr desarrollar competencias que les permitan consolidarse en el campo laboral, entre las cuales se encuentran la construcción de algoritmos, diseño de diagramas de flujo y desarrollo de programas en C++. En tal sentido, los estudiantes de ingeniería de sistemas, informática o carreras similares, deben preocuparse por sembrar las bases que les permitan fortalecerse; sobre todo, en lo relacionado con la programación; actividad medular dentro de su perfil profesional. Esta investigación pretende determinar el nivel de competencias específicas alcanzado por un grupo de estudiantes de esta disciplina, lo cual pudiera interpretarse como una situación que se presenta en poblaciones similares.

**Palabras clave:** competencias específicas, programación, nivel.

\* Ingeniero en Informática (URBE, 2005), Magíster Scientiarum. en Gerencia de Recursos Humanos (URBE, 2008). Docente de la Universidad del Zulia. Núcleo COL. Departamento de Ciencias Formales

\*\* Ingeniero en Sistemas (IUPSM, Extensión COL-Cabimas, 1999). Magíster Scientiarum en Informática Educativa, (URBE, 2010). Doctor en Ciencias de la Educación (URBE, 2012). Docente de la Universidad del Zulia. Núcleo COL. Departamento de Ciencias Formales.

## *Specific Skills in Programming Students in Santiago Mariño Polytechnic Institute*

---

### **Abstract**

Due to the permanent changes in the economic, political, and social environment that everyone lives in the country, our professionals should endeavor to be more competitive. For this, it is necessary to be able to develop skills that allow them to consolidate in the workplace, among which are the construction of algorithms, flowcharts design and the development of C ++ programming. Thus, system engineering, computer science, or similar career students should be concerned with setting the foundations that will let them stronger performances; especially in programming which constitute the core activities in their professional profile. This research intends to determine the specific skill levels achieved by the control group, which could be considered a standard measurement parameter.

**Key words:** specific skills, programming, level.

### **Introducción**

A través de los años, la humanidad ha evolucionado. La inventiva, el razonamiento y la lógica han dado lugar a los avances tecnológicos. El fenómeno de la globalización ha permitido su desarrollo en todo el mundo y, por ende, se ha dado solución a situaciones que hasta hace pocos años, se realizaban utilizando recursos manuales. Esto ha significado, tanto a los países desarrollados como subdesarrollados, la preocupación por crear sus propios productos tecnológicos, entre los cuales, está el software dirigido a la solución de problemas.

En virtud de lo anteriormente expuesto, se han creado carreras dirigidas a la formación de recurso humano orientado hacia la creación de productos tecnológicos desde la perspectiva del software, surgiendo así, los analistas de sistemas, programadores, desarrolladores de software o profesiones similares, las cuales deben cumplir una serie de características o competencias que les permitan construir sistemas de información, software de aplicación y de propósito específico, en búsqueda de resolver problemas existentes en las organizaciones tanto públicas como privadas.

En ese orden de ideas, Alles (2012) define las competencias como aquellas características subyacentes de la persona, relacionadas con una

correcta actuación en su puesto de trabajo y que pueden basarse en la motivación, en los rasgos de carácter, en actitudes o valores, en una variedad de conocimientos o capacidades cognoscitivas o de conducta.

Como complemento a lo anterior, la UNESCO (2008) en lo referente a la formación de competencias en tecnologías de la información y la comunicación (TIC) en los estudiantes, destaca que en general, las instituciones educativas en los distintos niveles, deben preocuparse por formar recurso humano competente, de acuerdo a las exigencias de los mercados de trabajo y con pertinencia suficiente dentro de su contexto.

Entonces, en el caso de Venezuela, se han producido avances en la reformulación de los planes de estudio, cambiando el paradigma basado en objetivos hacia el enfoque de competencias, para, Inciarte y Canquiz (2008) las instituciones de educación superior, deben formar profesionales competitivos, desde las distintas dimensiones del aprendizaje: el ser, conocer, hacer, y convivir; creando un individuo integral, capaz de utilizar las tecnologías de forma crítica y ética.

A la luz de lo expuesto anteriormente, se puede afirmar que los conocimientos específicos, se deben desarrollar en los estudiantes considerando sus competencias, para formar un ser humano no sólo capaz de acumular conocimientos en un área en particular, sino que pueda aplicarlo de forma práctica y manifiesta en su vida profesional. Ahora bien, el instituto universitario politécnico Santiago Mariño, extensión Costa Oriental del Lago-Cabimas, se dedica a la formación de profesionales en el área de ingeniería, específicamente de la especialidad en sistemas, capacitando nuevos profesionales dedicados a la automatización de procesos dentro de las organizaciones, desarrollando software de aplicación, dirigido a la solución de problemas y optimización de tareas repetitivas.

Por tal motivo, resulta importante indagar acerca de las competencias de los estudiantes iniciados en el ámbito de la programación, a efectos de determinar si efectivamente son alcanzadas al culminar las unidades curriculares orientadas al desarrollo de habilidades en cuanto al manejo de las técnicas básicas de fabricación de software. Por todo lo referido anteriormente, se plantea la siguiente interrogante: ¿Cuál es el nivel de desarrollo de competencias específicas en el área de programación de los estudiantes del instituto universitario politécnico Santiago Mariño, extensión COL-Cabimas?

Para dar respuesta a esta interrogante, se plantea como objetivo general determinar las competencias específicas alcanzadas por los estudiantes de programación de la carrera ingeniería de sistemas en el instituto universitario politécnico Santiago Mariño, extensión COL-Cabimas. Para ello, se plantea diagnosticar los conocimientos previos en el área de programación en los referidos estudiantes, describir las competencias específicas que deberán alcanzar y medir el nivel de alcance de las mismas al culminar la asignatura lenguaje de programación I.

## **Fundamentación teórica**

Una competencia puede definirse como el grado de dominio de las habilidades para realizar una actividad; siendo que los conocimientos se aplican en la vida cotidiana y existe la pericia necesaria para afrontar un trabajo, (Calvo, 2009). En ese orden de ideas, existen distintas formas de clasificar las competencias, principalmente Tobón (2010), quien habla de las específicas, las cuales, tienen un alto grado de especialización e implican procesos educativos específicos llevados a cabo en programas técnicos de formación para el trabajo y educación superior.

Seguidamente, se presenta la postura de Inciarte y Canquiz (2008), quienes consideran la definición del perfil profesional como un conjunto de competencias, los cuales poseen dimensiones cognoscitivas, procedimentales y actitudinales de la profesión y el nivel educativo. Este perfil orienta la relación de integralidad, coherencia, consistencia con los demás componentes del diseño curricular, incluyendo los perfiles de ingreso, docente e institucional.

Consecuentemente, los referidos autores afirman que las competencias cognoscitivas comprenden al área específica del saber, es decir, todos los conceptos que los estudiantes deben aprender y estén estrechamente relacionados con su área de desempeño profesional. Los estudiantes pueden preparar si se parte de los conocimientos previos, sobre todo cuando se relacionan con los “nuevos”, en virtud de afianzarlos y formar su propio andamiaje cognitivo.

Por su parte, las procedimentales comprenden de acuerdo a los aportes de Escamilla (2008) un conjunto de acciones que facilitan el logro de un fin propuesto. El estudiante es protagonista en la ejecución de cada uno de los pasos requeridos para construir su conocimiento; en otras palabras, desarrollará su capacidad para hacer.

En tal sentido, para Colina y Díaz (2013) las actitudinales están referidas a los valores, normas, creencias, actitudes conducentes al equilibrio personal y a la convivencia social. Como se pudo apreciar la actitud es considerada como una propiedad individual que define el comportamiento humano relacionado directamente con el ser y están relacionadas con la adquisición de conocimientos y con las experiencias que presenten modelos a partir de los cuales los estudiantes pueden reflexionar.

## Metodología

En función de los objetivos planteados inicialmente, esta investigación fue descriptiva, de campo y prospectiva; según Hernández *et al* (2014) quienes aseguran que la investigación de este tipo busca especificar propiedades, características y rasgos importantes de cualquier fenómeno sometido al análisis.

Al respecto, la presente investigación corresponde al tipo descriptivo, pues, en ella se desarrollan cada una de las dimensiones asociadas a la variable competencias específicas; indicando para este caso, las habilidades que los estudiantes deben desarrollar al culminar la asignatura Lenguaje de Programación I.

De igual manera, se cataloga como un estudio de campo, de acuerdo al criterio de Arias (2012) quien lo define como la recolección de datos tomados directamente de la realidad donde ocurren los hechos sin la manipulación de variable alguna. En este sentido, el estudio es de campo, y por consiguiente, la información se recopila directamente de la realidad donde se encuentra el objeto de estudio; es decir, el instituto universitario politécnico Santiago Mariño, ubicado en la ciudad de Cabimas.

Por consiguiente, al considerar el tiempo de recolección de la información, el estudio se clasifica como prospectivo, desde la perspectiva de Arias (2012), según su opinión, los estudios prospectivos son aquellos donde la información se recolecta de acuerdo a los criterios del investigador; previa planeación con fines específicos.

La investigación según su diseño, se clasifica como no experimental, transeccional, Según Hernández *et al*. (2014), este tipo de diseños recolectan datos en un solo momento; en un tiempo único. Su propósito es describir variables y su incidencia o interrelación en un momento deter-

minado. Igualmente, en este tipo de diseños no se manipulan deliberadamente las variables.

En virtud de lo anterior, se debe acotar que en el presente estudio no existe ningún tipo de manipulación de las variables, y la aplicación del instrumento de recolección de datos se realiza en un solo momento durante el transcurso de la investigación. A efectos del presente estudio, se tomó una población limitada para la aplicación del instrumento de recolección de datos, tal como se indican en los siguientes criterios de selección:

Está conformada por estudiantes de la asignatura *Lenguaje de Programación I* de la carrera Ingeniería de Sistemas en el Instituto Universitario Politécnico Santiago Mariño, extensión COL- Cabimas, turnos matutino y nocturno. En total son 15 personas. Debido a la accesibilidad de la misma, se tomó como muestra la distribución de la población de acuerdo a los planteamientos anteriores, se observa en la Tabla 1.

**Tabla 1.**  
Distribución de la población

	Características	Cantidad
Instituto Universitario Politécnico "Santiago Mariño" extensión COL-Cabimas	Estudiantes de la unidad curricular Lenguaje de programación I	15
Total de sujetos de la población		15

Elaboración propia (2015).

El instrumento utilizado para medir la variable en esta investigación, fue una prueba práctica, consistente en el planteamiento de un (01) problema donde el estudiante debía construir el algoritmo, realizar el diagrama de flujo y por último, codificar el programa en el lenguaje C para solucionar el mismo. El entorno de desarrollo y compilador utilizados fueron el Dev C++, el cual, funciona bajo entorno Windows y se encontraba instalado en los laboratorios de computación de la institución. Cada sección del instrumento se calificó separadamente del 01 al 20, pudiendo alcanzar puntajes diversos en cada una de ellas, ya que abarca las tres grandes unidades temáticas de la asignatura: construcción de algoritmos, diagramas de flujo y codificación en el lenguaje C++.

En lo concerniente al tiempo, se contó con un bloque de dos (02) horas académicas de cuarenta y cinco (45) minutos cada una, dejando

lapso de 30 minutos para la revisión por parte del docente; es decir, en efecto, los estudiantes contaron con 60 minutos en total para la realización de la prueba; destacando que se abarcaron las estructuras secuenciales y repetitivas vistas en clase durante el semestre; combinándolas dentro de un menú con opciones delimitadas por el docente.

Una vez diseñado el instrumento de evaluación, se aplicó la validación por parte de expertos en el área, seleccionándose para ello, un total de cinco (05) docentes de programación, con dominio de los contenidos abarcados en este estudio. Se constató entonces que el instrumento estuvo adecuado al nivel de principiante y el tiempo estipulado para la resolución, era suficiente; pudiendo obtener la máxima calificación en la escala cuantitativa del 01 al 20.

Consecuentemente; se estableció una escala de valoración para otorgar una cualificación adecuada dependiendo del puntaje obtenido en la escala referida anteriormente, quedando el baremo tal como se muestra en la siguiente Tabla:

**Tabla 2.**

Baremo de interpretación de resultados

Competencia	Rango	Significado
Construye algoritmos a partir de un problema planteado	20-19	Alto nivel de desempeño
	18-15	Buen nivel de desempeño
	14-10	Medio nivel de desempeño
	9-0	Deficiente nivel de desempeño
Diseña diagramas de flujo a partir de lo planteado en el algoritmo	20-19	Alto nivel de desempeño
	18-15	Buen nivel de desempeño
	14-10	Medio nivel de desempeño
	9-0	Deficiente nivel de desempeño
Desarrolla programas en C++ orientados a la solución de problemas planteados, utilizando las estructuras selectivas y repetitivas del lenguaje.	20-19	Alto nivel de desempeño
	18-15	Buen nivel de desempeño
	14-10	Medio nivel de desempeño
	9-0	Deficiente nivel de desempeño

Elaboración propia (2015).

## Resultados

Primeramente, al diagnosticar los conocimientos previos en los estudiantes, se encontró que en su mayoría (un total de 12) no habían tenido contacto con lenguajes de programación, aunque habían utilizado algoritmos en la solución de ejercicios matemáticos a nivel de educación media. Igualmente, desconocían el uso de los diagramas de flujo, su aplicación e importancia. Se percibió que de modo predominante, han utilizado el computador con fines de entretenimiento, socialización y elaboración de asignaciones escolares.

El resto de los estudiantes (03) en total, tenían nociones básicas sobre el lenguaje algorítmico y el uso de la simbología estandarizada para la construcción de diagramas de flujo, incluso, uno de ellos manifestó haber realizado en algún momento sencillos programas en lenguaje C. Toda esta información se recabó a partir de una exploración realizada durante los primeros encuentros en el aula, a través de la observación y la toma de notas.

Seguidamente, al describir las competencias específicas a ser alcanzados por los estudiantes de programación; específicamente en la referida unidad curricular, se tiene que de acuerdo al diseño existente en la institución objeto de estudio, la asignatura se divide en tres (03) unidades, las cuales son:

- a. Fundamentos de programación y *algoritmos*: El alumno debe ser capaz de analizar un problema cotidiano y expresarlo en lenguaje algorítmico; entendiéndose, de acuerdo al criterio de Llanos (2010), que son un conjunto de pasos organizados de manera lógica y secuencial para dar respuesta a un problema. En otras palabras, el estudiante debe conocer el problema, analizarlo y expresar mediante una secuencia de etapas sucesivas, las operaciones necesarias para resolverlo.
- b. Representación de algoritmos mediante diagramas de flujo: El alumno deberá representar un algoritmo de manera gráfica, utilizando los símbolos estandarizados para tal fin, siendo estos diagramas definidos por Joyanes (2008) como una representación gráfica de un proceso, donde cada paso corresponde a un símbolo, utilizando flechas para indicar la dirección del flujo y terminales para delimitar el inicio y fin del mismo.
- c. Introducción a la programación en C++: En esta unidad, el alumno deberá realizar programas en el lenguaje C++ utilizando estructuras básicas como variables, constantes, funciones estándar de lectura y escritura, ciclos, decisiones simples y múltiples; entre otras instrucciones nativas del referido lenguaje. En



tal sentido Schildt (2014) asevera que el C++, es un lenguaje de nivel medio, creado a partir del C estándar, con soporte para programación estructurada y orientada a objetos, que ha dado origen a numerosas herramientas de programación, convirtiéndose en un estándar en el mercado.

Igualmente, afirma que otros lenguajes como Java, PHP y Javascript tienen su base en el C estándar, incluso, algunos sistemas operativos como Unix/Linux están cimentados sobre la base de C; por lo cual, el desarrollo de habilidades como programador en esta área, representa una base sólida para el aprendizaje de sus lenguajes derivados.

Para dar cumplimiento al objetivo de medir el nivel de alcance de las competencias específicas por parte de los estudiantes, previamente descritos, se utilizó el baremo de interpretación presentado en la tabla 2, obteniéndose los siguientes resultados:

En primer lugar, al estudiar la competencia “construye algoritmos a partir de un problema planteado”, se encontró que un 42% de los estudiantes lograron un medio nivel de desempeño, mientras el 26% mostraron un buen nivel, un 12% no alcanzó desarrollar las habilidades esperadas en la utilización de los algoritmos y el resto (20%), evidenció un alto nivel en cuanto al alcance de la competencia esperada; por lo cual, se infiere que en general, los alumnos están en capacidad de analizar un problema y utilizar el lenguaje algorítmico para plantear los pasos que darán solución al mismo. Sin embargo, sólo un 20% demostró tener la capacidad para realizar la actividad sin ningún tipo de ayuda o errores en la ejecución.

Por esto, presumimos, los alumnos deben continuar trabajando en función de consolidar sus habilidades en la construcción de algoritmos mediante casos típicos, revisar contenidos tanto conceptuales como procedimentales y de esta manera, alcanzar el nivel óptimo esperado.

En cuanto a la competencia relacionada con el diseño de diagramas de flujo, los resultados mostraron que un 52% tuvo un desempeño medio, mientras un 28% mostró un bajo nivel de logro. Complementariamente, un 9% estuvo en la categoría superior (alto) y el resto (11%) se clasificó como “bueno”. De lo anterior se puede deducir que aun cuando los alumnos pueden desarrollar algoritmos, presentan problemas al utilizar la simbología estándar propia de los diagramas de flujo.

Esto puede notarse cuando poco más de la mitad de la muestra, tuvo fluctuaciones en cuanto a la elaboración de los mismos. Al realizar una re-

visión de los instrumentos de evaluación una vez aplicados, se pudo apreciar que algunos confundían los símbolos y el sentido que debía seguir el problema cuando se requería de una decisión o ciclo repetitivo. Por esto, se plantea, en el caso de los estudiantes, deben familiarizarse con los fundamentos conceptuales de las estructuras repetitivas y decisivas, su aplicación e instrumentación dentro de los programas de computadora.

Por último, en cuanto al desarrollo de programas en C++, sólo un 8% de los sujetos examinados presentó un nivel alto de desempeño; un 48% mostró un nivel medio y un 15% se catalogó como “bueno”. El resto de los estudiantes (29%) evidenciaron deficiencias en el uso del lenguaje. En virtud de lo anterior y al realizar una revisión minuciosa de los programas realizados por los alumnos, se pudo apreciar que presentaron problemas al momento de estructurar y codificar las instrucciones; pues las mismas presentaban errores tipográficos. Igualmente, se evidenciaron fallas en el dominio del idioma inglés; representando una barrera al momento de la escritura correcta de las instrucciones.

En el mismo orden de ideas, los sujetos mostraron debilidades al estructurar los ciclos y decisiones dentro del programa. Por estas razones, resulta conveniente, en el caso del docente, el tratar de manejar estrategias que ayuden al alumno a desarrollar la lógica de programación e incentivarlos al manejo del idioma inglés. Se sugiere la integración de software como inter-p, concebido con fines de enseñanza, el cual permite la escritura de programas sencillos en español. Los resultados referidos anteriormente, se muestran en la siguiente Tabla:

**Tabla 3.**  
 Resultados obtenidos

Competencia	%	Nivel
Construye algoritmos a partir de un problema planteado	20	Alto nivel de desempeño
	26	Buen nivel de desempeño
	42	Medio nivel de desempeño
	12	Deficiente nivel de desempeño
<b>Total</b>	<b>100%</b>	
Diseña diagramas de flujo a partir de lo planteado en el algoritmo	9	Alto nivel de desempeño
	11	Buen nivel de desempeño
	52	Medio nivel de desempeño
	28	Deficiente nivel de desempeño
<b>Total</b>	<b>100%</b>	

**Tabla 3 (Continuación)**

Competencia	%	Nivel
Desarrolla programas en C++ orientados a la solución de problemas planteados, utilizando las estructuras selectivas y repetitivas del lenguaje.	8	Alto nivel de desempeño
	15	Buen nivel de desempeño
	48	Medio nivel de desempeño
	29	Deficiente nivel de desempeño
Total	100%	

Elaboración propia (2015).

## Conclusiones

En primer lugar, se diagnosticaron los conocimientos previos en la población objeto de estudio; encontrándose debilidades en los fundamentos básicos de informática y programación. Cabe destacar que los sujetos egresaron recientemente del sistema educativo venezolano a nivel diversificado, lo cual hace presumir que aun cuando existen proyectos para alfabetizar tecnológicamente a la población, se mantienen algunas fluctuaciones en cuanto a la vinculación de esos conocimientos con la vida cotidiana; la consolidación de los mismos y la instrumentación práctica desde el punto de vista crítico-reflexivo para la resolución de problemas. Por tanto, se sugiere incluir unidades curriculares orientadas a introducir al estudiante en el lenguaje algorítmico y desarrollar programas de computación adecuados ante situaciones planteadas.

En segundo lugar, se describieron las competencias específicas a desarrollar por parte de los estudiantes de la asignatura Lenguaje de programación I en el Instituto universitario politécnico "Santiago Mariño", extensión COL-Cabimas, tales como la construcción de algoritmos, diseño de diagramas de flujo y desarrollo de programas en lenguaje C++. Se pudo observar la existencia de una sistematización o secuencia lógica dentro de los contenidos contemplados en el diseño curricular de la asignatura; sin embargo, deben instrumentarse estrategias y recursos que permitan al alumno afianzar los conocimientos conceptuales, para aplicarlos mediante un procedimiento y de esta manera, obtener los mejores resultados.

Finalmente, en cuanto al desarrollo de programas en el lenguaje C++, se pudo apreciar en los educandos, problemas al momento de aplicar la sintaxis del lenguaje, estructurar las instrucciones básicas de entrada/salida,

utilizar los ciclos repetitivos y de decisión. Se denotaron dificultades en el manejo del inglés y el pensamiento lógico-secuencial; así como desconocimiento de procedimientos sencillos de cálculo. Por lo anterior, se recomienda integrar al proceso educativo herramientas como el intérprete inter-p o P-seint; el cual, permiten ejecutar algoritmos escritos en idioma español, indicando los errores sintácticos y semánticos, según sea el caso, guiando así al estudiante para la resolución del problema planteado.

## Referencias bibliográficas

- Alles, Martha (2012). **Las 50 herramientas de recursos humanos que todo profesional debe conocer**. Editorial Granica, Buenos Aires, Argentina.
- Arias, Fidas (2012). **El Proyecto de Investigación. Introducción a la Metodología Científica**, Sexta edición. Editorial Epísteme. Caracas, Venezuela.
- Unesco (2008). **Estándares por competencias. TIC para docentes**. Londres, Inglaterra.
- Calvo, Luis (2009). **Competencias básicas. ¿Innovar o reinventar?** Editorial Serrano Villalba. España.
- Colina, Alicia y Díaz, Ángel (2013). **La formación de investigadores en educación y la producción del conocimiento**. Universidad Autónoma de Tlaxcala, Ediciones Díaz de Santos. Madrid, España.
- Escamilla, Amparo. (2008). **Las competencias básicas. Claves y propuestas para su desarrollo en los centros**. Editorial GRAO de IRIF, SL. Barcelona, España
- Hernández, Roberto; Fernández, Carlos y Baptista, Pilar. (2014). **Metodología de la Investigación**, Sexta edición. Editorial Mc. Graw Hill Education 2014. México DF, México
- Inciarte, Alicia y Canquiz, Liliana (2008). **Formación integral desde el enfoque por competencias**. Universidad del Zulia. Colección textos universitarios. Maracaibo, Venezuela.
- Joyanes, Luis (2008). **Fundamentos de programación**. Mc. Graw Hill Interamericana. Madrid, España.
- Llanos, Diego (2010). **Fundamentos de informática y programación en C**. Editorial Parainfo. Madrid, España.
- Schildt, Herbert (2014). **Java. The complete reference**. Novena. Edición. Mc. Graw Hill Education. USA
- Tobón, Sergio (2010). **Formación integral y competencias. Pensamiento complejo. Currículo, didáctica y Evaluación**. Ediciones ECOE. Bogotá, Colombia.