

Algoritmos desde la mirada de las Tecnologías de la Información y la Administración

1.1 ¿Qué es un algoritmo y por qué importa en administración?

Un **algoritmo** (*Algorithm*) es una secuencia finita, ordenada y precisa de pasos diseñada para resolver un problema o producir un resultado. En Tecnologías de la Información (TI), los algoritmos son la base de la programación, el procesamiento de datos, la automatización de tareas y muchas decisiones operativas que las organizaciones ejecutan diariamente.

Desde la administración, comprender algoritmos no significa aprender a programar. Significa entender cómo una organización transforma datos de entrada en resultados útiles mediante reglas, procedimientos y sistemas. Una liquidación de sueldos, una aprobación de crédito, una clasificación de clientes, el cálculo de un stock mínimo, la asignación de turnos o la detección de operaciones inusuales pueden depender de algoritmos simples o complejos.

Todo algoritmo trabaja con tres elementos:

Elemento	En inglés	Descripción
Entrada	<i>Input</i>	Los datos iniciales que el algoritmo necesita para operar
Proceso	<i>Process</i>	El conjunto de instrucciones que transforma esos datos
Salida	<i>Output</i>	El resultado producido

Esta estructura permite analizar tanto un programa informático como un procedimiento administrativo automatizado.

1.2 El algoritmo como procedimiento administrativo

Un algoritmo puede leerse como una rutina administrativa formalizada. La siguiente secuencia de pasos para recibir y procesar una factura puede implementarse como función dentro de un sistema de gestión:

1. Recibir la factura.
2. Verificar que el proveedor esté registrado.
3. Comparar la factura con la orden de compra.
4. Verificar la recepción de bienes o servicios.
5. Si los datos coinciden → enviar a aprobación.
6. Si los datos no coinciden → devolver para revisión.
7. Registrar el estado del trámite.

El valor de formalizar este procedimiento como algoritmo es que **cada caso se evalúa con la misma lógica**. Eso reduce la arbitrariedad, mejora la trazabilidad y facilita la auditoría.

Un algoritmo no reemplaza la gestión: organiza una decisión repetible y verificable.

1.3 Características de un buen algoritmo

Para ser útil dentro de un sistema de información, un algoritmo debe cumplir estas condiciones:

Característica	Qué implica	Consecuencia si falta
Precisión	Cada paso debe estar definido con claridad y sin ambigüedad	Una instrucción vaga como “revisar si el cliente es importante” produce resultados inconsistentes
Definición	Ante los mismos datos de entrada, produce el mismo resultado	Un cálculo impositivo o una liquidación que varían

Característica	Qué implica	Consecuencia si falta
		ante iguales datos es inauditable
Finitud	El algoritmo debe terminar	Un ciclo sin fin puede bloquear sistemas, consumir recursos y detener procesos programados
Legibilidad	Puede ser comprendido por quienes lo revisan	Sin legibilidad, el área usuaria no puede validar si la regla corresponde al proceso real
Eficiencia	Resuelve el problema con uso razonable de tiempo y recursos	Un algoritmo que tarda 8 horas en un proceso diario puede ser inadecuado si el resultado se necesita al inicio de la jornada
Correctitud	Resuelve el problema esperado, no solo ejecuta sin errores técnicos	Un algoritmo puede funcionar correctamente como programa y producir resultados incorrectos si la regla de negocio está mal definida

1.4 Algoritmos y reglas de negocio

Una **regla de negocio** (*Business Rule*) es una condición, restricción o criterio que define cómo debe actuar la organización ante una situación determinada. Muchas reglas de negocio se implementan mediante algoritmos.

Ejemplo: “todo descuento superior al 20% requiere aprobación gerencial”. Su traducción algorítmica:

1. Leer porcentaje de descuento solicitado.
2. Si el descuento es $\leq 20\%$ → aprobar según permisos del vendedor.
3. Si el descuento es $> 20\%$ → enviar a aprobación gerencial.
4. Registrar usuario, fecha, porcentaje y decisión.

La ventaja de implementar reglas mediante algoritmos es la **consistencia**: todos los casos se evalúan con el mismo criterio. La desventaja aparece cuando la regla está mal definida, no se actualiza o no contempla casos relevantes.

1.4.1 El error de automatizar un proceso desordenado

Un error frecuente es automatizar un proceso que todavía tiene ambigüedades. Si el procedimiento manual es inconsistente, la automatización repite esa inconsistencia a mayor velocidad.

Antes de programar conviene: - Revisar el proceso completo. - Definir responsables por etapa. - Establecer controles y criterios de excepción. - Documentar casos atípicos.

Administración y TI deben trabajar de manera coordinada. El área técnica puede programar la regla, pero el área de negocio debe validarla.

1.5 Representación de algoritmos

Los algoritmos pueden expresarse de distintas maneras según el destinatario:

Representación	Descripción	Destinatario adecuado
Lenguaje natural	Pasos expresados con frases comunes	Usuarios no técnicos; útil para explicar el proceso general
Pseudocódigo	Forma intermedia entre lenguaje humano y	Desarrolladores y analistas funcionales

Representación	Descripción	Destinatario adecuado
	programación; describe condiciones y repeticiones sin depender de un lenguaje específico	
Diagrama de flujo <i>(Flowchart)</i>	Representación visual mediante símbolos; útil para decisiones, circuitos de aprobación y rutas de excepción	Audidores, usuarios administrativos, responsables de procesos

Ejemplo de pseudocódigo para reposición de stock:

SI stock_actual < punto_reposicion ENTONCES

 generar_sugerencia_compra

SINO

 mantener_estado_actual

FIN SI

Para auditoría, la documentación debe permitir verificar entradas, reglas, salidas, responsables y evidencias. La representación elegida importa menos que la completitud de esa documentación.

1.6 Algoritmos en sistemas de información organizacionales

Los sistemas de información utilizan algoritmos para ejecutar tareas repetitivas y producir resultados consistentes. El siguiente cuadro muestra aplicaciones concretas por área funcional:

Área	Uso del algoritmo	Ejemplo concreto
Finanzas	Cálculo de intereses	Aplicar tasa mensual del 4% sobre saldos vencidos
Compras	Reposición automática	Sugerir compra si el stock cae por debajo del punto de reposición
Ventas	Asignación de descuentos	Autorizar descuento según volumen y perfil del cliente
Recursos humanos	Liquidación de haberes	Calcular salario, descuentos, horas extra y retenciones
Logística	Priorización de entregas	Ordenar pedidos por fecha prometida y zona de reparto
Atención al cliente	Clasificación de reclamos	Priorizar incidentes con clientes activos y servicios críticos
Auditoría interna	Detección de excepciones	Listar operaciones fuera de horario o por encima de un umbral

Cuando un sistema sugiere, clasifica, calcula, alerta o bloquea, hay una lógica algorítmica detrás. Comprender esa lógica permite a la administración validar, controlar y mejorar sus procesos.

1.7 Algoritmos, datos y calidad de la información

Un algoritmo depende de sus datos de entrada. La **calidad de datos** (*Data Quality*) se refiere al grado en que los datos son correctos, completos, actualizados, consistentes y adecuados para el uso previsto.

Dato incorrecto	Consecuencia en el algoritmo
Domicilios incompletos en la base de clientes	La asignación logística puede fallar o generar entregas incorrectas
Stock inexistente registrado en el sistema	El algoritmo de reposición no genera la compra necesaria
Datos salariales mal cargados	La liquidación de haberes puede ser incorrecta
Proveedor duplicado	El sistema puede generar pagos duplicados o aplicar condiciones no previstas

1.7.1 Controles sobre datos de entrada

Control	Finalidad	Ejemplo
Validación de formato	Evitar datos inválidos	No aceptar fechas imposibles ni importes negativos
Campos obligatorios	Evitar registros incompletos	Exigir CUIT, domicilio y condición fiscal de proveedores

Control	Finalidad	Ejemplo
Rangos permitidos	Evitar valores fuera de criterio	No aceptar descuentos superiores al máximo autorizado
Doble revisión	Reducir errores en datos críticos	Validar cambios de cuenta bancaria con confirmación por canal independiente
Conciliación	Comparar información entre fuentes	Cotejar facturas, órdenes de compra y recepciones
Registro de cambios	Dejar evidencia auditable	Guardar usuario, fecha y valor anterior ante cualquier modificación

Un algoritmo bien diseñado puede producir malos resultados si trabaja con datos incorrectos. En gestión, no alcanza con preguntar qué calcula el sistema. También hay que preguntar con qué datos calcula.

1.8 Automatización — Cuándo sí y cuándo no

La **automatización** (*Automation*) consiste en ejecutar tareas con baja intervención manual. Los algoritmos son su base operativa.

1.8.1 Tareas adecuadas para automatizar

Son candidatas a automatización las tareas que reúnen estas condiciones:

Condición	Ejemplo
Repetitivas y medibles	Validación de campos en formularios
Con reglas estables y claras	Cálculo de intereses por mora

Condición	Ejemplo
Bajo impacto ante errores o con revisión humana posterior	Generación de alertas de stock

Ejemplos concretos: conciliaciones simples, reportes periódicos, clasificación de tickets, alertas de vencimientos, generación de comprobantes.

1.8.2 Decisiones que requieren control adicional

Situación	Riesgo de automatizar sin revisión
Alto impacto y baja frecuencia	Un error afecta pocas operaciones pero con consecuencias graves
Ambigüedad en la regla	El sistema toma decisiones que la organización no previó
Excepciones frecuentes	La regla general no cubre muchos casos reales

Ejemplo: rechazar automáticamente a un proveedor estratégico por un dato incompleto puede ser riesgoso si no existe un canal de revisión formal.

Si la regla está mal formulada, el error también se automatiza.

1.9 Algoritmos, control interno y auditoría

El control interno busca asegurar que las operaciones sean confiables, autorizadas, registradas y alineadas con las políticas de la organización. Los algoritmos pueden **fortalecer o debilitar** ese control.

1.9.1 Cuándo fortalecen el control interno

Mecanismo	Ejemplo
Aplican reglas consistentes sin variación arbitraria	Todos los descuentos se calculan con la misma lógica
Registran evidencia de cada decisión	Log de aprobación con usuario, fecha e importes
Reducen errores manuales en procesos repetitivos	Detección automática de pagos duplicados
Permiten detectar excepciones	Alerta por operaciones superiores a un umbral definido

1.9.2 Cuándo debilitan el control interno

Problema	Consecuencia
Reglas no documentadas	No se puede explicar por qué el sistema tomó una decisión
Reglas no probadas	Un error puede ejecutarse durante meses sin detectarse
Sin registro de cambios	No se puede saber quién modificó un parámetro crítico
Acceso no controlado a parámetros	Alguien puede cambiar límites de crédito, tasas o descuentos sin aprobación

1.9.3 Requisitos de control sobre algoritmos

Un algoritmo en producción debe cumplir los siguientes controles para ser **auditable** (*Auditability*):

Paso	Acción
1	Documentación de la regla en lenguaje comprensible para el área usuaria
2	Identificación del responsable del proceso
3	Evidencia de pruebas antes de implementar
4	Registro de cambios con usuario, fecha y versión anterior
5	Aprobación formal de modificaciones
6	Monitoreo de resultados en producción
7	Revisión periódica de excepciones

1.10 Riesgos asociados al uso de algoritmos

Riesgo	Descripción	Ejemplo
Error de diseño	La regla no representa correctamente el proceso	Calcular un descuento sobre precio final cuando debía calcularse sobre precio neto
Datos incorrectos	La entrada contiene errores que el algoritmo procesa sin cuestionar	Un cliente duplicado recibe beneficios no previstos
Falta de actualización	La regla queda desactualizada respecto a los cambios del negocio o la normativa	Un algoritmo impositivo mantiene una alícuota vigente años atrás

Riesgo	Descripción	Ejemplo
Ausencia de trazabilidad	No se puede explicar por qué el sistema tomó una decisión	Un rechazo de crédito sin registro de los datos ni la regla aplicada
Sesgo en los datos (<i>Bias</i>)	Los datos históricos contienen patrones que generan resultados injustificados o no deseados	Un sistema de clasificación de clientes que discrimina por zona geográfica sin justificación
Dependencia excesiva	Los usuarios aceptan el resultado sin revisar, incluso ante señales evidentes de error	Una orden de compra automática de 10.000 unidades ante un error de demanda
Acceso indebido a parámetros	Personas no autorizadas modifican valores críticos del algoritmo	Un usuario modifica el límite de crédito o la tasa de interés sin aprobación

La gestión de estos riesgos requiere controles técnicos y administrativos. No basta con que el sistema funcione: debe funcionar correctamente, con datos confiables, reglas aprobadas y resultados revisables.

1.11 Algoritmos y toma de decisiones

Los algoritmos pueden cumplir distintos roles en el proceso de toma de decisiones organizacionales:

Tipo	Qué hace	Ejemplo
Descriptivo	Organiza información sobre lo ocurrido	Calcular ventas mensuales por región

Tipo	Qué hace	Ejemplo
Diagnóstico	Identifica causas o patrones en los datos	Detectar que los reclamos aumentan en un producto o zona específica
Predictivo	Estima eventos futuros con base en datos históricos	Proyectar demanda para los próximos 30 días
Prescriptivo	Recomienda o ejecuta una acción	Sugerir una compra, asignar un recurso o priorizar una entrega

Principio de proporcionalidad: cuanto mayor sea el impacto de la decisión, mayor debe ser el nivel de control sobre el algoritmo que la sustenta.

Decisión automatizada	Nivel de control recomendado
Recomendación de stock revisada por el área de compras	Bajo — el humano valida antes de actuar
Bloqueo automático de una cuenta de cliente	Alto — requiere reglas claras, comunicación adecuada y mecanismo de revisión y apelación
Rechazo automático de crédito	Alto — impacta derechos, debe ser explicable y revisable
Suspensión automática de servicio a proveedor	Alto — riesgo operativo y contractual

1.12 Ejemplo integrador — Reposición de stock con algoritmo

Una empresa que vende productos físicos desea reducir faltantes de stock. Define el siguiente algoritmo:

1.12.1 Datos de entrada

Dato	Valor
Ventas promedio diarias	30 unidades
Plazo de entrega del proveedor	6 días
Stock de seguridad	80 unidades
Stock actual	210 unidades
Cantidad mínima de compra	100 unidades

1.12.2 Regla definida

punto_reposicion = (ventas_promedio_diarias × plazo_entrega_dias) + stock_seguridad

punto_reposicion = (30 × 6) + 80 = 260 unidades

SI stock_actual ≤ punto_reposicion ENTONCES

sugerir_compra (cantidad mínima = 100 unidades)

SINO

no_generar_compra

FIN SI

1.12.3 Resultado

Variable	Valor
Punto de reposición calculado	260 unidades
Stock actual	210 unidades
¿Stock actual ≤ punto de reposición?	Sí (210 ≤ 260)
Salida del sistema	Sugerir compra de 100 unidades

1.12.4 Controles administrativos sobre el algoritmo

Control	Descripción
Revisión semanal de ventas promedio	Verificar que el dato de demanda esté actualizado
Validación periódica de plazos de proveedor	Los plazos pueden cambiar; un dato viejo altera el punto de reposición
Aprobación de compras superiores a 500 unidades	Umbral de revisión humana para órdenes grandes
Registro de usuarios que modifican parámetros	Auditoría de cambios en datos críticos del algoritmo

Este ejemplo muestra el valor administrativo del algoritmo: no solo calcula, sino que **ordena datos, aplica una regla, produce una salida y deja una base para controlar.**

1.13 Ideas clave

- Un algoritmo es una secuencia finita, ordenada y precisa de pasos. En administración, equivale a **una regla de gestión implementada en un sistema**: si la regla está mal definida, el resultado será incorrecto aunque el programa funcione sin errores técnicos.
- La estructura **entrada-proceso-salida** aplica tanto a un programa informático como a un procedimiento administrativo automatizado. Identificar los tres elementos es el primer paso para analizar cualquier decisión algorítmica.
- Las seis características de un buen algoritmo — precisión, definición, finitud, legibilidad, eficiencia y correctitud — son también **requisitos de calidad del proceso** que el algoritmo representa. Una falla en cualquiera de ellas tiene consecuencias operativas concretas.

- La **regla de negocio** debe validarla el área usuaria, no solo diseñarla el área técnica. Un error de diseño automatizado a gran velocidad puede generar errores masivos en pagos, liquidaciones, stocks o reportes.
- La **calidad de los datos de entrada** determina la calidad de los resultados. Un algoritmo bien diseñado produce salidas incorrectas si trabaja con datos incorrectos. Los controles sobre datos de entrada son parte integral de la gestión del algoritmo.
- La **automatización** no elimina el riesgo: lo desplaza. Si la regla está mal formulada, el error también se automatiza. Las decisiones de alto impacto requieren mecanismos de revisión humana, incluso cuando el proceso es automático.
- Los algoritmos **fortalecen el control interno** cuando están documentados, probados, registran evidencia y detectan excepciones. Lo **debilitan** cuando sus reglas son opacas, no tienen trazabilidad o sus parámetros pueden modificarse sin control.
- Los administradores deben poder **dialogar con las áreas técnicas**, validar reglas de negocio, interpretar salidas del sistema, pedir evidencia, detectar excepciones y evaluar si los resultados son razonables. En organizaciones dependientes de sistemas, eso es parte de la gestión.

1.14 Preguntas de evaluación

1. ¿Qué es un algoritmo y por qué es importante para la administración de organizaciones que utilizan sistemas de información?
2. Explique la estructura entrada-proceso-salida con un ejemplo aplicado al proceso de liquidación de haberes o al circuito de pagos a proveedores.
3. ¿Por qué un algoritmo puede ejecutarse sin errores técnicos y, aun así, producir resultados incorrectos? ¿Qué condición se viola en ese caso?

4. ¿Qué relación existe entre una regla de negocio y un algoritmo? ¿Por qué esa relación requiere coordinación entre el área de administración y el área técnica?
5. ¿Qué riesgos aparecen cuando se automatiza un proceso que aún tiene ambigüedades en su procedimiento manual?
6. ¿Cómo puede la mala calidad de los datos de entrada afectar el resultado de un algoritmo? Proporcione dos ejemplos concretos de áreas funcionales distintas.
7. ¿En qué tipo de decisiones la automatización requiere mayor nivel de control humano y por qué? Diferencie entre decisiones descriptivas, predictivas y prescriptivas.
8. ¿Qué controles deberían aplicarse antes y después de modificar un algoritmo que calcula descuentos, límites de crédito o puntos de reposición de stock?
9. ¿Por qué la ausencia de trazabilidad en un algoritmo representa un problema para el control interno y para la auditoría?
10. En el ejemplo integrador de reposición de stock: ¿qué ocurre si el dato de “ventas promedio diarias” está desactualizado porque hubo una temporada alta? ¿Qué control administrativo permite detectar ese problema?

1.15 Glosario

Término	Traducción / Explicación
Algorithm	Algoritmo. Secuencia finita, ordenada y precisa de pasos diseñada para resolver un problema o producir un resultado.
Auditability	Auditabilidad. Capacidad de una decisión de ser reconstruida y explicada con base en datos, reglas y registros documentados.
Automation	Automatización. Ejecución de tareas con baja o nula intervención manual, mediante reglas predefinidas y sistemas que las aplican de forma consistente.

Término	Traducción / Explicación
Bias	Sesgo. En algoritmos, tendencia en los datos históricos o en las reglas que genera resultados sistemáticamente injustificados o no deseados, frecuentemente en sistemas de clasificación o scoring.
Business Rule	Regla de negocio. Condición, restricción o criterio que define cómo debe actuar la organización ante una situación determinada. Muchas reglas de negocio se implementan mediante algoritmos.
CUIT	Clave Única de Identificación Tributaria. Identificador fiscal argentino asignado a personas jurídicas y físicas con actividad comercial.
Data Quality	Calidad de datos. Grado en que los datos son correctos, completos, actualizados, consistentes y adecuados para el uso previsto.
Flowchart	Diagrama de flujo. Representación visual de un algoritmo mediante símbolos estandarizados para pasos, decisiones, entradas y salidas.
Input	Entrada. Datos iniciales que el algoritmo necesita para operar. La calidad de los resultados depende directamente de la calidad de la entrada.
IS	<i>Information System.</i> Sistema de Información (SI). Combinación de personas, procesos, datos y tecnología orientada a producir información útil para la toma de decisiones.

Término	Traducción / Explicación
IT	<i>Information Technology</i> . Tecnologías de la Información (TI). Conjunto de recursos tecnológicos utilizados para procesar, almacenar, transmitir y proteger información.
Output	Salida. Resultado producido por el algoritmo a partir de los datos de entrada y las instrucciones del proceso.
Process	Proceso. Conjunto de instrucciones que transforma los datos de entrada en el resultado esperado.
Programming	Programación. Disciplina que consiste en escribir instrucciones que una computadora puede ejecutar para resolver un problema o automatizar una tarea.
Pseudocode	Pseudocódigo. Forma intermedia entre lenguaje natural y lenguaje de programación, utilizada para describir la lógica de un algoritmo sin depender de un lenguaje de programación específico.
Scoring	Puntuación o puntaje. Sistema algorítmico que asigna un valor numérico a una entidad (cliente, proveedor, operación) para clasificarla o priorizar decisiones sobre ella.