

Evaluación Final

A continuación, se define el instrumento de evaluación final del módulo:

Componentes de la evaluación	
Evidencias	Las evidencias de la evaluación serán <ul style="list-style-type: none">• De producto, consistente en:<ul style="list-style-type: none">○ Código Python con algoritmo• Conocimiento fundamental, que consiste en:<ul style="list-style-type: none">○ Hoja de respuesta
Situación Evaluativa	Se evaluará el desempeño mediante: <ul style="list-style-type: none">• Un encargo
Tipo de trabajo	Individual
Tiempo	2 horas
Lugar	Trabajo remoto

Enunciado del Problema:

Una empresa sospecha que sus correos están siendo interceptados por la competencia, por lo cual, decide desarrollar un algoritmo de encriptación. Un analista de datos que estuvo en el curso de Python propone el siguiente algoritmo:

En cada una de las palabras que compone el texto de entrada, tomar cada uno de los caracteres de la palabra y rotarla en 13 posiciones del abecedario. Es decir, si se toma como base el siguiente abecedario: ABCDEFGHIJKLMNOPQRSTUVWXYZ, entonces, el abecedario rotado será: NÑOPQRSTUVWXYZABCDEFGHIJKLM. Por lo tanto, si a nuestro algoritmo entra la palabra HOLA, entonces la salida debería ser TBXN.

Tome en consideración que los caracteres que no son reconocidos, por ejemplo, números, espacios, guiones y otros, no sufrirán modificaciones. Considere, además, que este algoritmo no debe ser sensible a mayúsculas y minúsculas.

Por ejemplo,

Hola -> Tbxn

HOLA! -> TBXN!

H4L1 -> T4X1

Hola como te va -> Tbxn oby b gq in

Requisitos del programa:

- Construya una función que reciba como parámetro una cadena de caracteres y retorne la cadena de caracteres encriptada
- Construya una función que reciba como parámetro una cadena de caracteres encriptada y retorne la cadena de caracteres desencriptada
- Escriba pruebas de ambas funciones en donde encripte y desencripte las cadenas de caracteres para verificar que las funciones realizan sus operaciones de forma correcta
- Debe considerar pruebas con todas las reglas y condiciones de bordes mencionadas

Criterios de evaluación:

Aspectos a evaluar	Descripción
Correctitud del algoritmo	El algoritmo implementado produce los resultados esperados y cumple con los requisitos del problema. Los resultados son precisos y están libres de errores.
Eficiencia y optimización	El algoritmo utiliza estructuras de datos eficientes y técnicas de optimización cuando sea necesario. La complejidad temporal y espacial es adecuada para el tamaño de entrada esperado.
Claridad y legibilidad del código	El código está bien estructurado y organizado, con nombres de variables y funciones descriptivos. Se siguen las convenciones de estilo de Python (PEP 8).
Modularidad y reutilización del código	El código se divide en funciones o clases lógicas, evitando la repetición innecesaria de código. Se promueve la reutilización del código en diferentes partes del programa.
Manejo de errores y excepciones	El código maneja adecuadamente los errores y excepciones esperados, utilizando estructuras de control de flujo para gestionar casos excepcionales.
Comentarios y documentación	El código está debidamente comentado para explicar el propósito y la lógica detrás de las secciones clave. Se proporciona documentación clara sobre el funcionamiento del algoritmo.
Pruebas y validación del algoritmo	Se incluyen pruebas exhaustivas del algoritmo para diferentes casos de entrada, demostrando su correcto funcionamiento y proporcionando una cobertura adecuada del código.
Estilo de programación y buenas prácticas	El código sigue las buenas prácticas de programación, evita malas prácticas y respeta las convenciones de estilo de Python.
Comportamiento y salida del programa	El programa se ejecuta sin errores y produce los resultados esperados. La salida es clara, legible y presenta la información adecuadamente.

