

Instituto Tecnológico y de Estudios Superiores de Monterrey

Escuela de Ingeniería

Maestría en Inteligencia Artificial Aplicada.

Pruebas de Software y aseguramiento de la calidad.



Actividad 4.2 Ejercicio de programación 1

David Elías Limas Palacios

A01560923

Fecha de entrega: 8 de febrero de 2026

Evidencia de ejecución y análisis estático

Programa 1

Análisis estático con Pylint

Se ejecutó el analizador estático pylint para verificar que el programa no presenta errores, advertencias ni problemas de calidad de código después de aplicar el estándar PEP-8 y corregir los detalles detectados.

- Comando ejecutado: *pylint computeStatistics.py*
- Resultado:

```
(.venv) PS C:\Users\david\OneDrive\Documentos\Documentos\Profesional y Master\Master\Materias\Pruebas de SW y QA\Tarea 4.2 Ejercicios de programacion 1\A01560923_A4.2> cd P1
(.venv) PS C:\Users\david\OneDrive\Documentos\Documentos\Profesional y Master\Master\Materias\Pruebas de SW y QA\Tarea 4.2 Ejercicios de programacion 1\A01560923_A4.2\P1> pylint co
mputeStatistics.py
-----
Your code has been rated at 10.00/10 (previous run: 10.00/10, +0.00)
```

Ejecución de casos de prueba

Se ejecutaron todos los casos de prueba proporcionados para validar la correcta funcionalidad del programa.

- Se utilizó el comando: *python computeStatistics.py TC[X].txt*
 - Donde [X] corresponde al número de caso de prueba.
 - Se corrió este comando para 7 casos de prueba diferentes.
- A la vez, se utilizó el comando: *python ComputeStatistics.py TC[X].txt > TC[X]_resultado.txt* para generar un archivo de texto para cada caso de prueba que funcione como evidencia de los resultados arrojados por el programa.
- Este comando generó un reporte diferente para cada uno de los casos de prueba, los cuales fueron consolidados posteriormente en un solo reporte llamado *ComputeStatistics_TC_resultados.txt*

Programa 2

Análisis estático con Pylint

Se ejecutó el analizador estático pylint para verificar que el programa no presenta errores, advertencias ni problemas de calidad de código después de aplicar el estándar PEP-8 y corregir los detalles detectados.

- Comando ejecutado: *pylint convertnumbers.py*
- Resultado:

```
(.venv) PS C:\Users\david\OneDrive\Documentos\Documentos\Profesional y Master\Master\Materias\Pruebas de SW y QA\Tarea 4.2 Ejercicios de programacion 1\A01560923_A4.2> cd P2
(.venv) PS C:\Users\david\OneDrive\Documentos\Documentos\Profesional y Master\Master\Materias\Pruebas de SW y QA\Tarea 4.2 Ejercicios de programacion 1\A01560923_A4.2\P2> pylint convertNumbers.py
-----
Your code has been rated at 10.00/10 (previous run: 10.00/10, +0.00)
```

Ejecución de casos de prueba

Se ejecutaron todos los casos de prueba proporcionados para validar la correcta funcionalidad del programa.

- Se utilizó el comando: *python convertNumbers.py TC[X].txt*
 - Donde [X] corresponde al número de caso de prueba.
 - Se corrió este comando para 4 casos de prueba diferentes.
- A la vez, se utilizó el comando: *python convertNumbers.py TC[X].txt > TC[X]_resultado.txt* para generar un archivo de texto para cada caso de prueba que funcione como evidencia de los resultados arrojados por el programa.
- Este comando generó un reporte diferente para cada uno de los casos de prueba, los cuales fueron consolidados posteriormente en un solo reporte llamado *convertNumbers_TC_resultados.txt*

Programa 3

Análisis estático con Pylint

Se ejecutó el analizador estático pylint para verificar que el programa no presenta errores, advertencias ni problemas de calidad de código después de aplicar el estándar PEP-8 y corregir los detalles detectados.

- Comando ejecutado: *pylint wordCount.py*
- Resultado:

```
Terminal Local x

(.venv) PS C:\Users\david\OneDrive\Documentos\Documentos\Profesional y Master\Master\Materias\Pruebas de SW y QA\Tarea 4.2 Ejercicios de programacion 1\A01560923_A4.2> cd P3
(.venv) PS C:\Users\david\OneDrive\Documentos\Documentos\Profesional y Master\Master\Materias\Pruebas de SW y QA\Tarea 4.2 Ejercicios de programacion 1\A01560923_A4.2\P3> pylint wordCount.py
-----
Your code has been rated at 10.00/10 (previous run: 10.00/10, +0.00)

(.venv) PS C:\Users\david\OneDrive\Documentos\Documentos\Profesional y Master\Master\Materias\Pruebas de SW y QA\Tarea 4.2 Ejercicios de programacion 1\A01560923_A4.2\P3>
```

Ejecución de casos de prueba

Se ejecutaron todos los casos de prueba proporcionados para validar la correcta funcionalidad del programa.

- Se utilizó el comando: *python wordCount.py TC[X].txt*

- Donde [X] corresponde al número de caso de prueba.
 - Se corrió este comando para 5 casos de prueba diferentes.
- A la vez, se utilizó el comando: `python wordCount.py TC[X].txt > TC[X]_resultado.txt` para generar un archivo de texto para cada caso de prueba que funcione como evidencia de los resultados arrojados por el programa.
- Este comando generó un reporte diferente para cada uno de los casos de prueba, los cuales fueron consolidados posteriormente en un solo reporte llamado `wordCount_TC_resultados.txt`

Resultados

Se ejecutaron correctamente los tres programas desarrollados utilizando todos los archivos de prueba proporcionados por la materia. Para cada programa se corrieron los casos de prueba indicados y se generaron los reportes de salida correspondientes desde la consola. Posteriormente, los resultados obtenidos por los programas fueron comparados directamente con los resultados esperados provistos en los archivos de apoyo de la actividad. En todos los casos, los valores calculados coincidieron con los resultados esperados, por lo que se determinó que los programas cumplen con los requerimientos funcionales establecidos y que los casos de prueba fueron aprobados satisfactoriamente.

Adicionalmente, se ejecutó el análisis estático mediante la herramienta pylint y se corrigieron los problemas detectados hasta no presentar errores ni advertencias de calidad de código.

En consecuencia, considerando tanto la correcta ejecución de los casos de prueba como la validación mediante análisis estático, se concluye que los programas funcionan adecuadamente y satisfacen los requisitos solicitados en la actividad.

Los reportes de casos de prueba consolidados para cada programa, se encuentran en el repositorio como evidencia.