

# Software Testing and QA

## Activity 4.2 Report

Luis Sebastián Gan Cadena

A01186635

### Notes

Pylint was implemented using VSCode extension.

### Program 1

This is the first version of the code with the pylint notes:

computeStatistics.py 3

P1 > computeStatistics.py > ...

```
1 import sys
2 import time
3
4 '''
5 File version before pylint
6 '''
7
8 def read_numbers_from_file(filename):
9     """Reads numbers from a file, handling invalid data."""
10    numbers = []
11    with open(filename, 'r') as file:
12        for line in file:
13            try:
14                numbers.append(float(line.strip()))
15            except ValueError:
16                print(f"Warning: Invalid data skipped -> {line.strip()}")
17    return numbers
18
19 def compute_mean(data):
20     """Computes mean using basic sum and division."""
21     return sum(data) / len(data) if data else 0
22
23 def compute_median(data):
24     """Computes median by sorting and finding the middle value."""
25     n = len(data)
```

PROBLEMS 10 OUTPUT DEBUG CONSOLE TERMINAL PORTS Filter (e.g. text, \*\*/\*.ts, !\*\*/no

computeStatistics.py P1 10

- ⚠ Using open without explicitly specifying an encoding Pylint(W1514:unspecified-encoding) [Ln 7, Col 10]
- ⚠ Using open without explicitly specifying an encoding Pylint(W1514:unspecified-encoding) [Ln 50, Col 10]
- ⚠ Catching too general exception Exception Pylint(W0718:broad-exception-caught) [Ln 83, Col 12]
- ❗ Missing module docstring Pylint(C0114:missing-module-docstring) [Ln 1, Col 1]
- ❗ Module name "computeStatistics" doesn't conform to snake\_case naming style Pylint(C0103:invalid-name) [Ln 1, Col 1]
- ❗ Trailing whitespace Pylint(C0303:trailing-whitespace) [Ln 58, Col 1]
- ❗ Trailing whitespace Pylint(C0303:trailing-whitespace) [Ln 63, Col 1]
- ❗ Trailing whitespace Pylint(C0303:trailing-whitespace) [Ln 67, Col 1]
- ❗ Trailing whitespace Pylint(C0303:trailing-whitespace) [Ln 73, Col 1]

```

1
2 def write_results_to_file(results, elapsed_time):
3     """Writes statistics results to an output file."""
4     with open('StatisticsResults.txt', 'w') as file:
5         file.write(results + f"\nExecution Time: {elapsed_time:.4f} seconds\n")
6
7 def main():
8     """Main function to handle file reading, computation, and output."""
9     if len(sys.argv) != 2:
10         print("Usage: python computeStatistics.py <fileWithData.txt>")
11         sys.exit(1)
12
13     filename = sys.argv[1]
14     try:
15         start_time = time.time()
16         data = read_numbers_from_file(filename)
17
18         if not data:
19             print("Error: No valid numbers found in the file.")
20             sys.exit(1)
21
22         mean = compute_mean(data)
23         median = compute_median(data)
24         mode = compute_mode(data)
25         variance = compute_variance(data, mean)

```

```

26         mode = compute_mode(data)
27         variance = compute_variance(data, mean)
28         std_dev = compute_std_dev(variance)
29
30         elapsed_time = time.time() - start_time
31         results = (f"Mean: {mean}\nMedian: {median}\nMode: {mode}\n"
32                  | f"Variance: {variance}\nStandard Deviation: {std_dev}")
33         print(results + f"\nExecution Time: {elapsed_time:.4f} seconds")
34         write_results_to_file(results, elapsed_time)
35
36     except FileNotFoundError:
37         print(f"Error: File '{filename}' not found.")
38         sys.exit(1)
39     except Exception as e:
40         print(f"Unexpected error: {e}")
41         sys.exit(1)
42
43 if __name__ == "__main__":
44     main()

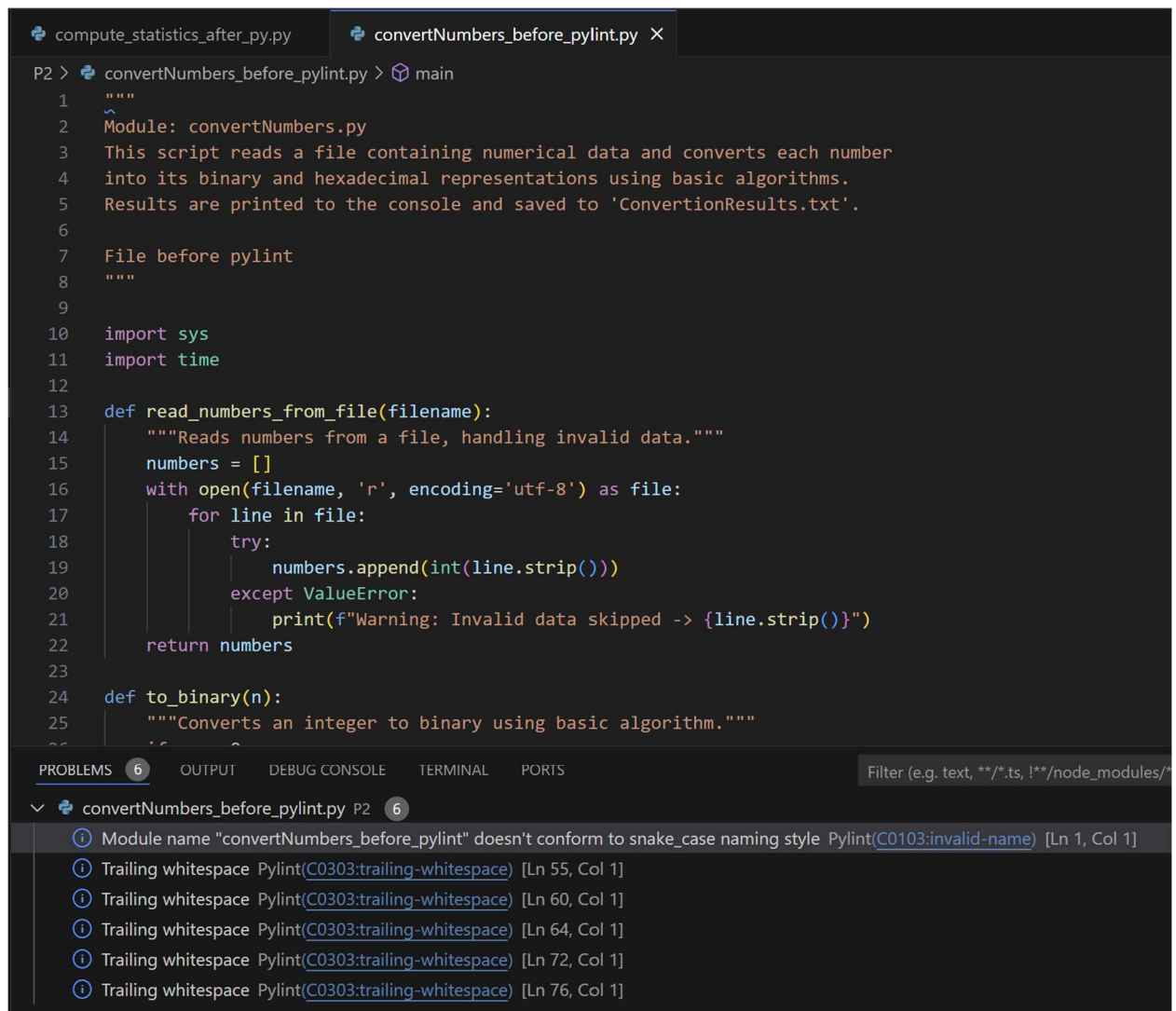
```

The notes were fixed and the result can be seen on the `compute_statistics_after_pylint.py` file.

The results of the program can be seen on the `StatisticsResults.txt` file.

## Program 2

For this program, using the retro from the first one not many issues were found with pylint, but the file naming and some spaces.



The screenshot shows a code editor with two tabs: `compute_statistics_after_py.py` and `convertNumbers_before_pylint.py`. The active tab is `convertNumbers_before_pylint.py`, which contains the following Python code:

```
1 """
2 Module: convertNumbers.py
3 This script reads a file containing numerical data and converts each number
4 into its binary and hexadecimal representations using basic algorithms.
5 Results are printed to the console and saved to 'ConversionResults.txt'.
6
7 File before pylint
8 """
9
10 import sys
11 import time
12
13 def read_numbers_from_file(filename):
14     """Reads numbers from a file, handling invalid data."""
15     numbers = []
16     with open(filename, 'r', encoding='utf-8') as file:
17         for line in file:
18             try:
19                 numbers.append(int(line.strip()))
20             except ValueError:
21                 print(f"Warning: Invalid data skipped -> {line.strip()}")
22     return numbers
23
24 def to_binary(n):
25     """Converts an integer to binary using basic algorithm."""
```

Below the code editor, the **PROBLEMS** panel shows 6 errors for `convertNumbers_before_pylint.py`:

- ① Module name "convertNumbers\_before\_pylint" doesn't conform to snake\_case naming style Pylint(C0103:invalid-name) [Ln 1, Col 1]
- ① Trailing whitespace Pylint(C0303:trailing-whitespace) [Ln 55, Col 1]
- ① Trailing whitespace Pylint(C0303:trailing-whitespace) [Ln 60, Col 1]
- ① Trailing whitespace Pylint(C0303:trailing-whitespace) [Ln 64, Col 1]
- ① Trailing whitespace Pylint(C0303:trailing-whitespace) [Ln 72, Col 1]
- ① Trailing whitespace Pylint(C0303:trailing-whitespace) [Ln 76, Col 1]

## Program 3

Now after the retro of the other programs we only get notification about some spaces again, the file name and the length of one code line.

Programs

wordCount\_before\_pylint.py X word\_count\_after\_pylint.py

P3 > wordCount\_before\_pylint.py > ...

```
1 """
2 Module: wordCount.py
3 This script reads a file containing words, identifies distinct words, and calculates their frequen
4 using basic algorithms. Results are printed to the console and saved to 'WordCountResults.txt'.
5
6 File before pylint
7 """
8
9 import sys
10 import time
11
12 def read_words_from_file(filename):
13     """Reads words from a file, handling invalid data."""
14     words = []
15     try:
16         with open(filename, 'r', encoding='utf-8') as file:
17             for line in file:
18                 words.extend(line.strip().split())
19     except FileNotFoundError:
20         print(f"Error: File '{filename}' not found.")
21         sys.exit(1)
22     except (OSError, IOError) as e:
23         print(f"Unexpected error: {e}")
24         sys.exit(1)
25     return words
```

PROBLEMS 13 OUTPUT DEBUG CONSOLE TERMINAL PORTS Filter (e.g. text, \*\*/\*.ts, !\*\*/node\_modu

> word\_count\_after\_pylint.py P3 6

wordCount\_before\_pylint.py P3 7

- Module name "wordCount\_before\_pylint" doesn't conform to snake\_case naming style Pylint(C0103:invalid-name) [Ln 1, Col 1]
- Line too long (102/100) Pylint(C0301:line-too-long) [Ln 3, Col 1]
- Trailing whitespace Pylint(C0303:trailing-whitespace) [Ln 48, Col 1]
- Trailing whitespace Pylint(C0303:trailing-whitespace) [Ln 52, Col 1]
- Trailing whitespace Pylint(C0303:trailing-whitespace) [Ln 56, Col 1]
- Trailing whitespace Pylint(C0303:trailing-whitespace) [Ln 58, Col 1]
- Trailing whitespace Pylint(C0303:trailing-whitespace) [Ln 63, Col 1]