**Software Engineering:**

**Exploring Software Metrics with Python**

Samantha M. Hipple

School of Technology and Computing, City University of Seattle

CS 504: Software Engineering

Kendra Schraml

June 6, 2022

**Software Engineering:**

**Exploring Software Metrics with Python**

Basic info about software metrics – brief history, general types, importance.

Software metrics are used to measure software characteristics that are quantifiable. [source](https://stackify.com/track-software-metrics/)

As a way of demonstrating the importance of metric testing, we will be using the metrics we discover to improve our code as we progress through the report.

There are four functions related to software metrics: (1) planning, (2) organizing, (3) controlling, and (4) improving. [source](https://www.geeksforgeeks.org/software-measurement-and-metrics/)

Software metrics are used to measure performance, plan working items, measure productivity, estimating costs, and more. The goal of tracking these metrics is to determine the quality of the software product, then apply what is learned to improve the system and predict its end-state quality. Software development managers often use software metrics to identify areas of improvement, manage workloads, and reduce costs while development team members can use them to better communicate that status of their projects, to find and address issues, and better manage their own workflow. [source](https://stackify.com/track-software-metrics/)

The first attempts at applying this idea of quantifying software characteristics to track software quality and development progress appeared in the 1970s. These initial metrics focused on coding design and attribute more so than a program’s overall architecture. TEXTBOOK

**Getting Started with Pylama**

This report explores software metric testing using three Python modules: **pylama**, **pylint**, and **radon**. Pylama is a code audit tool that wraps nine software metric tools in one: **pycodestyle**, **pydocstyle**, **pyflakes**, **mccabe**, **pylint**, **radon**, **eradicate**, **mypy**, and **vulture**. We began our metrics testing by running all of the metrics tools available in Pylama against the four modules of our Tic Tac Toe program: *main.py*, *ttt\_menus.py*, *tic\_tac\_toe.py*, and *test.py*. and saved the results to *report.txt* - which can be reviewed in Appendix A.

This initial Pylama report is sorted first by our program modules, then by the line number of the code being flagged. The tool used to generate each flag is recorded at the end of each message as well. In order to break down these results further, we also generated individual reports for each of the metric tools available in Pylama. In doing this, we discovered that the **vulture** module, which is designed to find unused and untested code; and the **eradicate** module, which is designed to remove all dead code (commented-out code) from your source code; return no analysis for our example program (nor make any changes). Additionally, the **mypy** module, which is a static type-checker that requires your code to be written using static-typing, is unable to analyze our Tic Tac Toe program as well, due to its standard of dynamic-typing throughout.

**Further Metrics Exploration with Radon**

Next, we used the **radon** module itself to provide a more in-depth look at our source code’s using its four software metric tools: raw metrics, cyclomatic complexity (CC), Halstead metrics, and maintainability index (MI) ([source](https://radon.readthedocs.io/en/latest/index.html)). This section will break down each of the four tools initial results by providing both detailed messages that are similar to the Pylama reports in Appendix A, as well as quantitative data structured into tables by Radon’s report feature. Additionally, we will compare the results of Radon’s CC tool to that of Pylama’s CC results.

**Raw Metrics:**

Radon uses the following data within its raw metrics reports: total lines of code (**LOC**) – which is equivalent to (**SLOC + Multi + Single comments + Blank**); logical lines of code (**LLOC**) – which includes all code instructions; and source lines of code (**SLOC**) – which include all lines of code, minus any comments and blank lines. Next, Radon records: the number of comment lines (**Comments**); the total number of comments that are not multi-line strings (**Single comments**); the total number of comments that are multi-line strings (**Multi**); and the total number of blank lines (**Blank**). Lastly, Radon uses the above metrics to compute three comment statistics: percentage of comments per total lines of code (**C % L**); percentage of total comments per lines of source code (**C % S**); and percentage of comments and multi-line strings per total lines of code (**M + C % L**) ([source](https://radon.readthedocs.io/en/latest/intro.html#raw-metrics)).

Looking at the results displayed in Figure 1 on the next page, we can see that our tic\_tac\_toe module contains the most comments per lines of code at 45%. Additionally, we can see that our ttt\_menus module contains zero comments – which has the potential to make the code more difficult to maintain or be understood by others. Our main module could potentially use more comments as well, while our test module appears well documented.

These raw metrics are able to establish common patterns that a programmer may not realize exits within their coding. One clear pattern from the developer of our example program is that of commenting larger modules well, while essentially ignoring comment use in simpler, shorter modules. Fixing this unintentional oversight would improve this programmer’s ability to share, maintain, and refractor their code more readily.

**Figure 1.** Initial Radon Raw MetricsA screenshot of a computer

Description automatically generated with low confidence

**Halstead Metrics:**

The Halstead Complexity metric uses four metric units when analyzing a program’s source code: (1) number of *distinct* operators, (2) number of *distinct* operands, (3) sum of all *operators*, (4) and the sum of all *operands*. Halstead then applied these values to define two more measurements: (1) unique program**Text

Description automatically generated** vocabulary (sum of unique operands and operators), and (2) program length (sum of all operands and operators) (source == textbook). From there, Radon takes the generated Halstead metrics and creates six more statistics for each module: (1) calculated length, (2) volume, (3) difficulty, (4) effort, (5) time, and (6) bugs.

The calculated length represents Halstead’s first hypothesis of software science - that the length of a well-structured program can be determined solely using the number of unique operators and operands in this formula: **N^ = n1log2n1 + n2log2n2**. The idea is to compare the actual and estimated lengths as a way of measuring whether or not your program is well-structured, however, this particular hypothesis is often inaccurate for smaller programs.

**Figure 2.** Initial Radon Halstead Metrics

**McCabe’s Cyclomatic Complexity**

McCabe directly related the complexity of a program’s control flow to its overall quality. That is, the number of branches in the program’s design or source code and is based on graph theory.

**Maintainability Index**

**Pylint**

The second metrics module for Python we delved into is called **pylint**.

“Most of the messages here will be self-explanatory, but the first letter in each line will map to Convention, Refactor, Warning, or Error — and with regards to coding style, Pylint follows the [PEP8 style guide](http://www.python.org/dev/peps/pep-0008/).” [source](https://blog.codacy.com/which-python-static-analysis-tools-should-i-use/)

**Appendix A**

Initial Pylama Results for All Modules

Namespace(paths=['C:/Users/Saman/Desktop/CS 504 (software engineering)/05. Independent Project/CS504-Project-Samantha-Hipple/Submission 4/TicTacToe v2.0'], verbose=True, options=None, linters=['pylint', 'mypy', 'mccabe', 'pycodestyle', 'pyflakes', 'pydocstyle', 'isort'], from\_stdin=False, concurrent=False, format='pycodestyle', abspath=False, max\_line\_length=100, select=set(), ignore=set(), skip=[], sort=None, report='report.txt', hook=False, max\_complexity=10, pydocstyle\_convention=None, pylint\_confidence=None, radon\_no\_assert=False, radon\_show\_closures=False, file\_params={}, linters\_params={})

Run [pylint] main.py

Run [mypy] main.py

Run [mccabe] main.py

Run [pycodestyle] main.py

Run [pyflakes] main.py

Run [pydocstyle] main.py

Run [isort] main.py

Run [pylint] test.py

Run [mypy] test.py

Run [mccabe] test.py

Run [pycodestyle] test.py

Run [pyflakes] test.py

Run [pydocstyle] test.py

Run [isort] test.py

Run [pylint] tic\_tac\_toe.py

Run [mypy] tic\_tac\_toe.py

Run [mccabe] tic\_tac\_toe.py

Run [pycodestyle] tic\_tac\_toe.py

Run [pyflakes] tic\_tac\_toe.py

Run [pydocstyle] tic\_tac\_toe.py

Run [isort] tic\_tac\_toe.py

Run [pylint] ttt\_menus.py

Run [mypy] ttt\_menus.py

Run [mccabe] ttt\_menus.py

Run [pycodestyle] ttt\_menus.py

Run [pyflakes] ttt\_menus.py

Run [pydocstyle] ttt\_menus.py

Run [isort] ttt\_menus.py

main.py:0:1 Incorrectly sorted imports. [isort]

main.py:1:1 C0114 Missing module docstring [pylint]

main.py:1:1 D100 Missing docstring in public module [pydocstyle]

main.py:8:1 C0116 Missing function or method docstring [pylint]

main.py:8:1 E302 expected 2 blank lines, found 1 [pycodestyle]

main.py:8:1 D103 Missing docstring in public function [pydocstyle]

main.py:19:32 C0303 Trailing whitespace [pylint]

main.py:19:1 C0325 Unnecessary parens after 'if' keyword [pylint]

main.py:19:13 R1723 Unnecessary "elif" after "break", remove the leading "el" from "elif" [pylint]

main.py:19:32 W291 trailing whitespace [pycodestyle]

main.py:22:34 C0303 Trailing whitespace [pylint]

main.py:22:1 C0325 Unnecessary parens after 'elif' keyword [pylint]

main.py:22:34 W291 trailing whitespace [pycodestyle]

main.py:25:1 C0325 Unnecessary parens after 'elif' keyword [pylint]

main.py:27:17 R1722 Consider using sys.exit() [pylint]

main.py:29:1 C0303 Trailing whitespace [pylint]

main.py:29:1 W293 blank line contains whitespace [pycodestyle]

main.py:30:1 C0116 Missing function or method docstring [pylint]

main.py:30:1 E302 expected 2 blank lines, found 1 [pycodestyle]

main.py:30:1 D103 Missing docstring in public function [pydocstyle]

main.py:39:13 R1722 Consider using sys.exit() [pylint]

main.py:41:32 C0303 Trailing whitespace [pylint]

main.py:41:1 C0325 Unnecessary parens after 'if' keyword [pylint]

main.py:41:32 W291 trailing whitespace [pycodestyle]

main.py:43:1 C0325 Unnecessary parens after 'if' keyword [pylint]

main.py:45:17 R1722 Consider using sys.exit() [pylint]

main.py:48:1 E305 expected 2 blank lines after class or function definition, found 1 [pycodestyle]

main.py:50:1 C0304 Final newline missing [pylint]

main.py:50:12 W292 no newline at end of file [pycodestyle]

test.py:0:1 Incorrectly sorted imports. [isort]

test.py:1:1 C0114 Missing module docstring [pylint]

test.py:1:1 D100 Missing docstring in public module [pydocstyle]

test.py:4:1 C0411 standard import "from io import StringIO" should be placed before "import numpy as np" [pylint]

test.py:5:1 C0411 standard import "from unittest.mock import patch" should be placed before "import numpy as np" [pylint]

test.py:6:1 C0411 third party import "import pytest" should be placed before "from tic\_tac\_toe import TicTacToe" [pylint]

test.py:6:1 W0611 Unused import pytest [pylint]

test.py:12:1 E302 expected 2 blank lines, found 1 [pycodestyle]

test.py:13:1 D203 1 blank line required before class docstring [pydocstyle]

test.py:13:1 D300 Use """triple double quotes""" [pydocstyle]

test.py:18:27 E261 at least two spaces before inline comment [pycodestyle]

test.py:20:48 E261 at least two spaces before inline comment [pycodestyle]

test.py:28:78 C0303 Trailing whitespace [pylint]

test.py:28:78 W291 trailing whitespace [pycodestyle]

test.py:28:32 E261 at least two spaces before inline comment [pycodestyle]

test.py:32:23 E222 multiple spaces after operator [pycodestyle]

test.py:33:24 E222 multiple spaces after operator [pycodestyle]

test.py:34:1 C0301 Line too long (115/100) [pylint]

test.py:35:24 E222 multiple spaces after operator [pycodestyle]

test.py:36:24 E222 multiple spaces after operator [pycodestyle]

test.py:37:24 E222 multiple spaces after operator [pycodestyle]

test.py:38:1 C0301 Line too long (115/100) [pylint]

test.py:39:24 E222 multiple spaces after operator [pycodestyle]

test.py:40:24 E222 multiple spaces after operator [pycodestyle]

test.py:41:24 E222 multiple spaces after operator [pycodestyle]

test.py:42:1 C0301 Line too long (115/100) [pylint]

test.py:43:24 E222 multiple spaces after operator [pycodestyle]

test.py:44:24 E222 multiple spaces after operator [pycodestyle]

test.py:50:1 D400 First line should end with a period [pydocstyle]

test.py:50:1 D415 First line should end with a period, question mark, or exclamation point [pydocstyle]

test.py:51:26 E261 at least two spaces before inline comment [pycodestyle]

test.py:52:24 E261 at least two spaces before inline comment [pycodestyle]

test.py:56:53 E251 unexpected spaces around keyword / parameter equals [pycodestyle]

test.py:59:1 C0301 Line too long (102/100) [pylint]

test.py:60:41 E261 at least two spaces before inline comment [pycodestyle]

test.py:61:37 E261 at least two spaces before inline comment [pycodestyle]

test.py:62:45 E261 at least two spaces before inline comment [pycodestyle]

test.py:64:41 E261 at least two spaces before inline comment [pycodestyle]

test.py:65:37 E261 at least two spaces before inline comment [pycodestyle]

test.py:66:45 E261 at least two spaces before inline comment [pycodestyle]

test.py:68:53 E251 unexpected spaces around keyword / parameter equals [pycodestyle]

test.py:69:31 C0303 Trailing whitespace [pylint]

test.py:69:31 W291 trailing whitespace [pycodestyle]

test.py:72:41 E261 at least two spaces before inline comment [pycodestyle]

test.py:73:68 E261 at least two spaces before inline comment [pycodestyle]

test.py:74:57 E261 at least two spaces before inline comment [pycodestyle]

test.py:75:46 E261 at least two spaces before inline comment [pycodestyle]

test.py:77:41 E261 at least two spaces before inline comment [pycodestyle]

test.py:78:68 E261 at least two spaces before inline comment [pycodestyle]

test.py:79:57 E261 at least two spaces before inline comment [pycodestyle]

test.py:80:46 E261 at least two spaces before inline comment [pycodestyle]

test.py:82:27 C0303 Trailing whitespace [pylint]

test.py:82:27 W291 trailing whitespace [pycodestyle]

test.py:84:1 D300 Use """triple double quotes""" [pydocstyle]

test.py:85:41 E261 at least two spaces before inline comment [pycodestyle]

test.py:86:35 E261 at least two spaces before inline comment [pycodestyle]

test.py:87:1 C0301 Line too long (103/100) [pylint]

test.py:87:57 E261 at least two spaces before inline comment [pycodestyle]

test.py:89:53 E251 unexpected spaces around keyword / parameter equals [pycodestyle]

test.py:94:45 E261 at least two spaces before inline comment [pycodestyle]

test.py:95:25 E261 at least two spaces before inline comment [pycodestyle]

test.py:96:45 E261 at least two spaces before inline comment [pycodestyle]

test.py:97:52 E261 at least two spaces before inline comment [pycodestyle]

test.py:98:38 E261 at least two spaces before inline comment [pycodestyle]

test.py:99:65 E261 at least two spaces before inline comment [pycodestyle]

test.py:101:53 E251 unexpected spaces around keyword / parameter equals [pycodestyle]

test.py:106:45 E261 at least two spaces before inline comment [pycodestyle]

test.py:107:25 E261 at least two spaces before inline comment [pycodestyle]

test.py:108:45 E261 at least two spaces before inline comment [pycodestyle]

test.py:109:44 E261 at least two spaces before inline comment [pycodestyle]

test.py:110:49 E261 at least two spaces before inline comment [pycodestyle]

test.py:111:65 E261 at least two spaces before inline comment [pycodestyle]

test.py:121:5 E265 block comment should start with '# ' [pycodestyle]

test.py:124:36 C0303 Trailing whitespace [pylint]

test.py:124:36 W291 trailing whitespace [pycodestyle]

test.py:128:1 E305 expected 2 blank lines after class or function definition, found 1 [pycodestyle]

test.py:129:1 C0304 Final newline missing [pylint]

test.py:129:21 W292 no newline at end of file [pycodestyle]

test.py:129:20 W291 trailing whitespace [pycodestyle]

tic\_tac\_toe.py:0:1 Incorrectly sorted imports. [isort]

tic\_tac\_toe.py:1:1 C0114 Missing module docstring [pylint]

tic\_tac\_toe.py:1:1 D100 Missing docstring in public module [pydocstyle]

tic\_tac\_toe.py:2:1 C0411 standard import "import random" should be placed before "import numpy as np" [pylint]

tic\_tac\_toe.py:3:1 C0411 standard import "from time import sleep" should be placed before "import numpy as np" [pylint]

tic\_tac\_toe.py:6:1 E302 expected 2 blank lines, found 1 [pycodestyle]

tic\_tac\_toe.py:7:1 D203 1 blank line required before class docstring [pydocstyle]

tic\_tac\_toe.py:7:1 D204 1 blank line required after class docstring [pydocstyle]

tic\_tac\_toe.py:7:1 D400 First line should end with a period [pydocstyle]

tic\_tac\_toe.py:9:1 D107 Missing docstring in \_\_init\_\_ [pydocstyle]

tic\_tac\_toe.py:10:24 E261 at least two spaces before inline comment [pycodestyle]

tic\_tac\_toe.py:11:68 E261 at least two spaces before inline comment [pycodestyle]

tic\_tac\_toe.py:12:1 C0301 Line too long (113/100) [pylint]

tic\_tac\_toe.py:12:47 E261 at least two spaces before inline comment [pycodestyle]

tic\_tac\_toe.py:15:5 C0116 Missing function or method docstring [pylint]

tic\_tac\_toe.py:15:1 D102 Missing docstring in public method [pydocstyle]

tic\_tac\_toe.py:16:47 E261 at least two spaces before inline comment [pycodestyle]

tic\_tac\_toe.py:17:33 E261 at least two spaces before inline comment [pycodestyle]

tic\_tac\_toe.py:18:52 E261 at least two spaces before inline comment [pycodestyle]

tic\_tac\_toe.py:21:5 C0116 Missing function or method docstring [pylint]

tic\_tac\_toe.py:21:1 D102 Missing docstring in public method [pydocstyle]

tic\_tac\_toe.py:25:29 E251 unexpected spaces around keyword / parameter equals [pycodestyle]

tic\_tac\_toe.py:27:46 E251 unexpected spaces around keyword / parameter equals [pycodestyle]

tic\_tac\_toe.py:30:55 C0303 Trailing whitespace [pylint]

tic\_tac\_toe.py:30:55 W291 trailing whitespace [pycodestyle]

tic\_tac\_toe.py:33:5 C0116 Missing function or method docstring [pylint]

tic\_tac\_toe.py:33:1 D102 Missing docstring in public method [pydocstyle]

tic\_tac\_toe.py:34:24 E261 at least two spaces before inline comment [pycodestyle]

tic\_tac\_toe.py:37:5 C0116 Missing function or method docstring [pylint]

tic\_tac\_toe.py:37:1 D102 Missing docstring in public method [pydocstyle]

tic\_tac\_toe.py:38:21 E261 at least two spaces before inline comment [pycodestyle]

tic\_tac\_toe.py:39:16 R1714 Consider merging these comparisons with "in" to "marker not in ('X', 'O')" [pylint]

tic\_tac\_toe.py:39:49 E261 at least two spaces before inline comment [pycodestyle]

tic\_tac\_toe.py:40:71 E261 at least two spaces before inline comment [pycodestyle]

tic\_tac\_toe.py:41:34 C0303 Trailing whitespace [pylint]

tic\_tac\_toe.py:41:9 R1705 Unnecessary "else" after "return", remove the "else" and de-indent the code inside it [pylint]

tic\_tac\_toe.py:41:34 W291 trailing whitespace [pycodestyle]

tic\_tac\_toe.py:41:26 E261 at least two spaces before inline comment [pycodestyle]

tic\_tac\_toe.py:42:30 E261 at least two spaces before inline comment [pycodestyle]

tic\_tac\_toe.py:43:22 C0303 Trailing whitespace [pylint]

tic\_tac\_toe.py:43:22 W291 trailing whitespace [pycodestyle]

tic\_tac\_toe.py:43:14 E261 at least two spaces before inline comment [pycodestyle]

tic\_tac\_toe.py:44:30 E261 at least two spaces before inline comment [pycodestyle]

tic\_tac\_toe.py:47:5 C0116 Missing function or method docstring [pylint]

tic\_tac\_toe.py:47:30 E261 at least two spaces before inline comment [pycodestyle]

tic\_tac\_toe.py:47:1 D102 Missing docstring in public method [pydocstyle]

tic\_tac\_toe.py:48:96 C0303 Trailing whitespace [pylint]

tic\_tac\_toe.py:48:96 W291 trailing whitespace [pycodestyle]

tic\_tac\_toe.py:48:48 E261 at least two spaces before inline comment [pycodestyle]

tic\_tac\_toe.py:49:79 C0303 Trailing whitespace [pylint]

tic\_tac\_toe.py:49:79 W291 trailing whitespace [pycodestyle]

tic\_tac\_toe.py:49:32 E261 at least two spaces before inline comment [pycodestyle]

tic\_tac\_toe.py:52:5 C0116 Missing function or method docstring [pylint]

tic\_tac\_toe.py:52:25 E261 at least two spaces before inline comment [pycodestyle]

tic\_tac\_toe.py:52:1 D102 Missing docstring in public method [pydocstyle]

tic\_tac\_toe.py:53:36 E261 at least two spaces before inline comment [pycodestyle]

tic\_tac\_toe.py:56:5 C0116 Missing function or method docstring [pylint]

tic\_tac\_toe.py:56:1 D102 Missing docstring in public method [pydocstyle]

tic\_tac\_toe.py:57:1 C0301 Line too long (101/100) [pylint]

tic\_tac\_toe.py:58:47 E261 at least two spaces before inline comment [pycodestyle]

tic\_tac\_toe.py:59:20 E261 at least two spaces before inline comment [pycodestyle]

tic\_tac\_toe.py:62:5 C0116 Missing function or method docstring [pylint]

tic\_tac\_toe.py:62:1 D102 Missing docstring in public method [pydocstyle]

tic\_tac\_toe.py:63:40 E261 at least two spaces before inline comment [pycodestyle]

tic\_tac\_toe.py:64:20 E261 at least two spaces before inline comment [pycodestyle]

tic\_tac\_toe.py:65:46 E261 at least two spaces before inline comment [pycodestyle]

tic\_tac\_toe.py:66:22 E261 at least two spaces before inline comment [pycodestyle]

tic\_tac\_toe.py:69:5 C0116 Missing function or method docstring [pylint]

tic\_tac\_toe.py:69:1 D102 Missing docstring in public method [pydocstyle]

tic\_tac\_toe.py:70:28 E261 at least two spaces before inline comment [pycodestyle]

tic\_tac\_toe.py:71:31 E261 at least two spaces before inline comment [pycodestyle]

tic\_tac\_toe.py:72:66 C0303 Trailing whitespace [pylint]

tic\_tac\_toe.py:72:66 W291 trailing whitespace [pycodestyle]

tic\_tac\_toe.py:72:31 E261 at least two spaces before inline comment [pycodestyle]

tic\_tac\_toe.py:73:1 C0301 Line too long (102/100) [pylint]

tic\_tac\_toe.py:73:42 E261 at least two spaces before inline comment [pycodestyle]

tic\_tac\_toe.py:74:50 E261 at least two spaces before inline comment [pycodestyle]

tic\_tac\_toe.py:75:1 C0301 Line too long (102/100) [pylint]

tic\_tac\_toe.py:75:45 E261 at least two spaces before inline comment [pycodestyle]

tic\_tac\_toe.py:76:20 E261 at least two spaces before inline comment [pycodestyle]

tic\_tac\_toe.py:79:5 C0116 Missing function or method docstring [pylint]

tic\_tac\_toe.py:79:1 D102 Missing docstring in public method [pydocstyle]

tic\_tac\_toe.py:80:38 E261 at least two spaces before inline comment [pycodestyle]

tic\_tac\_toe.py:83:5 C0116 Missing function or method docstring [pylint]

tic\_tac\_toe.py:83:1 D102 Missing docstring in public method [pydocstyle]

tic\_tac\_toe.py:84:20 E261 at least two spaces before inline comment [pycodestyle]

tic\_tac\_toe.py:85:83 C0303 Trailing whitespace [pylint]

tic\_tac\_toe.py:85:83 W291 trailing whitespace [pycodestyle]

tic\_tac\_toe.py:85:17 E261 at least two spaces before inline comment [pycodestyle]

tic\_tac\_toe.py:86:49 E261 at least two spaces before inline comment [pycodestyle]

tic\_tac\_toe.py:87:58 E261 at least two spaces before inline comment [pycodestyle]

tic\_tac\_toe.py:88:38 E261 at least two spaces before inline comment [pycodestyle]

tic\_tac\_toe.py:89:66 C0303 Trailing whitespace [pylint]

tic\_tac\_toe.py:89:66 W291 trailing whitespace [pycodestyle]

tic\_tac\_toe.py:89:41 E261 at least two spaces before inline comment [pycodestyle]

tic\_tac\_toe.py:90:17 R1722 Consider using sys.exit() [pylint]

tic\_tac\_toe.py:90:23 E261 at least two spaces before inline comment [pycodestyle]

tic\_tac\_toe.py:91:13 W0702 No exception type(s) specified [pylint]

tic\_tac\_toe.py:91:13 E722 do not use bare 'except' [pycodestyle]

tic\_tac\_toe.py:91:20 E261 at least two spaces before inline comment [pycodestyle]

tic\_tac\_toe.py:92:53 E261 at least two spaces before inline comment [pycodestyle]

tic\_tac\_toe.py:93:18 E261 at least two spaces before inline comment [pycodestyle]

tic\_tac\_toe.py:94:44 E261 at least two spaces before inline comment [pycodestyle]

tic\_tac\_toe.py:95:22 W1514 Using open without explicitly specifying an encoding [pylint]

tic\_tac\_toe.py:95:62 E261 at least two spaces before inline comment [pycodestyle]

tic\_tac\_toe.py:96:86 C0303 Trailing whitespace [pylint]

tic\_tac\_toe.py:96:86 W291 trailing whitespace [pycodestyle]

tic\_tac\_toe.py:96:54 E261 at least two spaces before inline comment [pycodestyle]

tic\_tac\_toe.py:97:52 E261 at least two spaces before inline comment [pycodestyle]

tic\_tac\_toe.py:98:29 E261 at least two spaces before inline comment [pycodestyle]

tic\_tac\_toe.py:101:5 C0116 Missing function or method docstring [pylint]

tic\_tac\_toe.py:101:1 D102 Missing docstring in public method [pydocstyle]

tic\_tac\_toe.py:102:35 E261 at least two spaces before inline comment [pycodestyle]

tic\_tac\_toe.py:103:46 E261 at least two spaces before inline comment [pycodestyle]

tic\_tac\_toe.py:104:54 E261 at least two spaces before inline comment [pycodestyle]

tic\_tac\_toe.py:105:17 E261 at least two spaces before inline comment [pycodestyle]

tic\_tac\_toe.py:106:54 E261 at least two spaces before inline comment [pycodestyle]

tic\_tac\_toe.py:107:83 C0303 Trailing whitespace [pylint]

tic\_tac\_toe.py:107:14 W1514 Using open without explicitly specifying an encoding [pylint]

tic\_tac\_toe.py:107:83 W291 trailing whitespace [pycodestyle]

tic\_tac\_toe.py:107:54 E261 at least two spaces before inline comment [pycodestyle]

tic\_tac\_toe.py:108:46 E261 at least two spaces before inline comment [pycodestyle]

tic\_tac\_toe.py:109:44 E261 at least two spaces before inline comment [pycodestyle]

tic\_tac\_toe.py:112:5 C0116 Missing function or method docstring [pylint]

tic\_tac\_toe.py:112:1 D102 Missing docstring in public method [pydocstyle]

tic\_tac\_toe.py:113:60 C0303 Trailing whitespace [pylint]

tic\_tac\_toe.py:113:60 W291 trailing whitespace [pycodestyle]

tic\_tac\_toe.py:113:31 E261 at least two spaces before inline comment [pycodestyle]

tic\_tac\_toe.py:114:66 C0303 Trailing whitespace [pylint]

tic\_tac\_toe.py:114:66 W291 trailing whitespace [pycodestyle]

tic\_tac\_toe.py:114:31 E261 at least two spaces before inline comment [pycodestyle]

tic\_tac\_toe.py:115:1 C0301 Line too long (111/100) [pylint]

tic\_tac\_toe.py:115:1 C0325 Unnecessary parens after 'if' keyword [pylint]

tic\_tac\_toe.py:115:44 E261 at least two spaces before inline comment [pycodestyle]

tic\_tac\_toe.py:116:33 E261 at least two spaces before inline comment [pycodestyle]

tic\_tac\_toe.py:117:20 E261 at least two spaces before inline comment [pycodestyle]

tic\_tac\_toe.py:119:41 C0303 Trailing whitespace [pylint]

tic\_tac\_toe.py:119:41 W291 trailing whitespace [pycodestyle]

tic\_tac\_toe.py:120:5 C0116 Missing function or method docstring [pylint]

tic\_tac\_toe.py:120:5 R0912 Too many branches (14/12) [pylint]

tic\_tac\_toe.py:120:5 R1710 Either all return statements in a function should return an expression, or none of them should. [pylint]

tic\_tac\_toe.py:120:5 C901 'TicTacToe.is\_winner' is too complex (15) [mccabe]

tic\_tac\_toe.py:120:1 D102 Missing docstring in public method [pydocstyle]

tic\_tac\_toe.py:127:1 C0325 Unnecessary parens after 'if' keyword [pylint]

tic\_tac\_toe.py:128:1 C0325 Unnecessary parens after 'return' keyword [pylint]

tic\_tac\_toe.py:134:1 C0325 Unnecessary parens after 'if' keyword [pylint]

tic\_tac\_toe.py:135:1 C0325 Unnecessary parens after 'return' keyword [pylint]

tic\_tac\_toe.py:140:1 C0325 Unnecessary parens after 'if' keyword [pylint]

tic\_tac\_toe.py:141:1 C0325 Unnecessary parens after 'return' keyword [pylint]

tic\_tac\_toe.py:146:1 C0325 Unnecessary parens after 'if' keyword [pylint]

tic\_tac\_toe.py:147:1 C0325 Unnecessary parens after 'return' keyword [pylint]

tic\_tac\_toe.py:149:5 C0116 Missing function or method docstring [pylint]

tic\_tac\_toe.py:149:5 E301 expected 1 blank line, found 0 [pycodestyle]

tic\_tac\_toe.py:149:1 D102 Missing docstring in public method [pydocstyle]

tic\_tac\_toe.py:150:45 C0303 Trailing whitespace [pylint]

tic\_tac\_toe.py:150:45 W291 trailing whitespace [pycodestyle]

tic\_tac\_toe.py:151:55 C0303 Trailing whitespace [pylint]

tic\_tac\_toe.py:151:55 W291 trailing whitespace [pycodestyle]

tic\_tac\_toe.py:152:5 C0116 Missing function or method docstring [pylint]

tic\_tac\_toe.py:152:5 R1710 Either all return statements in a function should return an expression, or none of them should. [pylint]

tic\_tac\_toe.py:152:5 E301 expected 1 blank line, found 0 [pycodestyle]

tic\_tac\_toe.py:152:1 D102 Missing docstring in public method [pydocstyle]

tic\_tac\_toe.py:155:9 R1705 Unnecessary "elif" after "return", remove the leading "el" from "elif" [pylint]

tic\_tac\_toe.py:160:26 C0303 Trailing whitespace [pylint]

tic\_tac\_toe.py:160:26 W291 trailing whitespace [pycodestyle]

tic\_tac\_toe.py:163:5 C0103 Method name "game\_mode\_2\_PvE" doesn't conform to snake\_case naming style [pylint]

tic\_tac\_toe.py:164:1 D400 First line should end with a period [pydocstyle]

tic\_tac\_toe.py:164:1 D415 First line should end with a period, question mark, or exclamation point [pydocstyle]

tic\_tac\_toe.py:168:58 C0303 Trailing whitespace [pylint]

tic\_tac\_toe.py:168:18 W1514 Using open without explicitly specifying an encoding [pylint]

tic\_tac\_toe.py:168:58 W291 trailing whitespace [pycodestyle]

tic\_tac\_toe.py:169:30 W1309 Using an f-string that does not have any interpolated variables [pylint]

tic\_tac\_toe.py:169:30 f-string is missing placeholders [pyflakes]

tic\_tac\_toe.py:171:32 C0303 Trailing whitespace [pylint]

tic\_tac\_toe.py:171:32 W291 trailing whitespace [pycodestyle]

tic\_tac\_toe.py:173:53 C0303 Trailing whitespace [pylint]

tic\_tac\_toe.py:173:53 W291 trailing whitespace [pycodestyle]

tic\_tac\_toe.py:180:65 C0303 Trailing whitespace [pylint]

tic\_tac\_toe.py:180:65 W291 trailing whitespace [pycodestyle]

tic\_tac\_toe.py:196:24 C0303 Trailing whitespace [pylint]

tic\_tac\_toe.py:196:24 W291 trailing whitespace [pycodestyle]

tic\_tac\_toe.py:206:64 C0303 Trailing whitespace [pylint]

tic\_tac\_toe.py:206:64 W291 trailing whitespace [pycodestyle]

tic\_tac\_toe.py:211:22 C0303 Trailing whitespace [pylint]

tic\_tac\_toe.py:211:22 W291 trailing whitespace [pycodestyle]

tic\_tac\_toe.py:217:66 C0303 Trailing whitespace [pylint]

tic\_tac\_toe.py:217:26 W1514 Using open without explicitly specifying an encoding [pylint]

tic\_tac\_toe.py:217:66 W291 trailing whitespace [pycodestyle]

tic\_tac\_toe.py:220:26 C0303 Trailing whitespace [pylint]

tic\_tac\_toe.py:220:26 W291 trailing whitespace [pycodestyle]

tic\_tac\_toe.py:225:26 W1514 Using open without explicitly specifying an encoding [pylint]

tic\_tac\_toe.py:226:62 C0303 Trailing whitespace [pylint]

tic\_tac\_toe.py:226:62 W291 trailing whitespace [pycodestyle]

tic\_tac\_toe.py:228:26 C0303 Trailing whitespace [pylint]

tic\_tac\_toe.py:228:26 W291 trailing whitespace [pycodestyle]

tic\_tac\_toe.py:230:37 C0303 Trailing whitespace [pylint]

tic\_tac\_toe.py:230:37 W291 trailing whitespace [pycodestyle]

tic\_tac\_toe.py:238:16 C0121 Comparison 'self.replay() == False' should be 'self.replay() is False' if checking for the singleton value False, or 'not self.replay()' if testing for falsiness [pylint]

tic\_tac\_toe.py:238:30 E712 comparison to False should be 'if cond is False:' or 'if not cond:' [pycodestyle]

tic\_tac\_toe.py:241:1 C0303 Trailing whitespace [pylint]

tic\_tac\_toe.py:241:1 W293 blank line contains whitespace [pycodestyle]

tic\_tac\_toe.py:242:5 E303 too many blank lines (2) [pycodestyle]

tic\_tac\_toe.py:243:5 C0103 Method name "game\_mode\_1\_PvP" doesn't conform to snake\_case naming style [pylint]

tic\_tac\_toe.py:243:5 E301 expected 1 blank line, found 0 [pycodestyle]

tic\_tac\_toe.py:244:1 D400 First line should end with a period [pydocstyle]

tic\_tac\_toe.py:248:58 C0303 Trailing whitespace [pylint]

tic\_tac\_toe.py:248:18 W1514 Using open without explicitly specifying an encoding [pylint]

tic\_tac\_toe.py:248:58 W291 trailing whitespace [pycodestyle]

tic\_tac\_toe.py:249:30 W1309 Using an f-string that does not have any interpolated variables [pylint]

tic\_tac\_toe.py:249:30 f-string is missing placeholders [pyflakes]

tic\_tac\_toe.py:251:32 C0303 Trailing whitespace [pylint]

tic\_tac\_toe.py:251:32 W291 trailing whitespace [pycodestyle]

tic\_tac\_toe.py:253:53 C0303 Trailing whitespace [pylint]

tic\_tac\_toe.py:253:53 W291 trailing whitespace [pycodestyle]

tic\_tac\_toe.py:260:65 C0303 Trailing whitespace [pylint]

tic\_tac\_toe.py:260:65 W291 trailing whitespace [pycodestyle]

tic\_tac\_toe.py:273:49 C0303 Trailing whitespace [pylint]

tic\_tac\_toe.py:273:49 W291 trailing whitespace [pycodestyle]

tic\_tac\_toe.py:274:64 C0303 Trailing whitespace [pylint]

tic\_tac\_toe.py:274:64 W291 trailing whitespace [pycodestyle]

tic\_tac\_toe.py:277:37 C0303 Trailing whitespace [pylint]

tic\_tac\_toe.py:277:37 W291 trailing whitespace [pycodestyle]

tic\_tac\_toe.py:280:24 C0303 Trailing whitespace [pylint]

tic\_tac\_toe.py:280:24 W291 trailing whitespace [pycodestyle]

tic\_tac\_toe.py:285:66 C0303 Trailing whitespace [pylint]

tic\_tac\_toe.py:285:26 W1514 Using open without explicitly specifying an encoding [pylint]

tic\_tac\_toe.py:285:66 W291 trailing whitespace [pycodestyle]

tic\_tac\_toe.py:288:26 C0303 Trailing whitespace [pylint]

tic\_tac\_toe.py:288:26 W291 trailing whitespace [pycodestyle]

tic\_tac\_toe.py:293:26 W1514 Using open without explicitly specifying an encoding [pylint]

tic\_tac\_toe.py:294:62 C0303 Trailing whitespace [pylint]

tic\_tac\_toe.py:294:62 W291 trailing whitespace [pycodestyle]

tic\_tac\_toe.py:296:26 C0303 Trailing whitespace [pylint]

tic\_tac\_toe.py:296:26 W291 trailing whitespace [pycodestyle]

tic\_tac\_toe.py:298:37 C0303 Trailing whitespace [pylint]

tic\_tac\_toe.py:298:37 W291 trailing whitespace [pycodestyle]

tic\_tac\_toe.py:304:31 C0303 Trailing whitespace [pylint]

tic\_tac\_toe.py:304:31 W291 trailing whitespace [pycodestyle]

tic\_tac\_toe.py:306:16 C0121 Comparison 'self.replay() == False' should be 'self.replay() is False' if checking for the singleton value False, or 'not self.replay()' if testing for falsiness [pylint]

tic\_tac\_toe.py:306:30 E712 comparison to False should be 'if cond is False:' or 'if not cond:' [pycodestyle]

ttt\_menus.py:1:1 C0114 Missing module docstring [pylint]

ttt\_menus.py:1:1 C0115 Missing class docstring [pylint]

ttt\_menus.py:1:1 D100 Missing docstring in public module [pydocstyle]

ttt\_menus.py:1:1 D101 Missing docstring in public class [pydocstyle]

ttt\_menus.py:2:1 D107 Missing docstring in \_\_init\_\_ [pydocstyle]

ttt\_menus.py:3:13 C0303 Trailing whitespace [pylint]

ttt\_menus.py:3:13 W291 trailing whitespace [pycodestyle]

ttt\_menus.py:5:5 C0116 Missing function or method docstring [pylint]

ttt\_menus.py:5:1 D102 Missing docstring in public method [pydocstyle]

ttt\_menus.py:6:18 E222 multiple spaces after operator [pycodestyle]

ttt\_menus.py:13:78 C0303 Trailing whitespace [pylint]

ttt\_menus.py:13:78 W291 trailing whitespace [pycodestyle]

ttt\_menus.py:15:78 C0303 Trailing whitespace [pylint]

ttt\_menus.py:15:78 W291 trailing whitespace [pycodestyle]

ttt\_menus.py:23:1 C0325 Unnecessary parens after 'return' keyword [pylint]

ttt\_menus.py:25:5 C0116 Missing function or method docstring [pylint]

ttt\_menus.py:25:1 D102 Missing docstring in public method [pydocstyle]

ttt\_menus.py:28:5 C0116 Missing function or method docstring [pylint]

ttt\_menus.py:28:1 D102 Missing docstring in public method [pydocstyle]

ttt\_menus.py:31:1 C0325 Unnecessary parens after 'return' keyword [pylint]

ttt\_menus.py:33:5 C0116 Missing function or method docstring [pylint]

ttt\_menus.py:33:1 D102 Missing docstring in public method [pydocstyle]

ttt\_menus.py:34:20 E222 multiple spaces after operator [pycodestyle]

ttt\_menus.py:41:80 C0303 Trailing whitespace [pylint]

ttt\_menus.py:41:80 W291 trailing whitespace [pycodestyle]

ttt\_menus.py:43:80 C0303 Trailing whitespace [pylint]

ttt\_menus.py:43:80 W291 trailing whitespace [pycodestyle]

ttt\_menus.py:47:80 C0303 Trailing whitespace [pylint]

ttt\_menus.py:47:80 W291 trailing whitespace [pycodestyle]

ttt\_menus.py:49:1 C0325 Unnecessary parens after 'return' keyword [pylint]

ttt\_menus.py:51:5 C0116 Missing function or method docstring [pylint]

ttt\_menus.py:51:1 D102 Missing docstring in public method [pydocstyle]

ttt\_menus.py:52:1 C0304 Final newline missing [pylint]

ttt\_menus.py:52:46 W292 no newline at end of file [pycodestyle]

**Appendix B**

Initial Radon McCabe Cyclomatic Complexity Results

**Figure B1.** Initial CC results for main module

Text

Description automatically generated**Figure B2.** Initial CC results for ttt\_menus module

Text

Description automatically generated**Figure B3.** Initial CC results for tic\_tac\_toe module

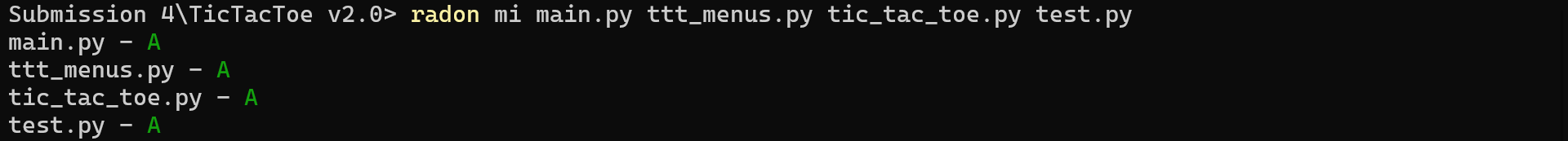
Text

Description automatically generated**Figure B4.** Initial CC results for test module

**Appendix C**

Text, letter

Description automatically generated**Appendix C.** Initial Pylama McCabe CC Results

**Figure 3.** Maintainability Index

**Pylint**

**A screenshot of a computer

Description automatically generated with low confidenceFigure #.** Initial Pylint report for main module

**Text

Description automatically generatedFigure #.** Initial Pylint results for main module

**Text, letter

Description automatically generatedFigure #.** Pylama’s initial Pylint results for main module

**Text

Description automatically generatedFigure #.** Initial Pylint results for ttt\_menus module

Text

Description automatically generated with low confidence**Figure #.** Pylama’s initial Pylint results for ttt\_menus module

**Graphical user interface

Description automatically generated with medium confidenceFigure #.** Initial Pylint report for ttt\_menus module

A screenshot of a computer

Description automatically generated with low confidence**Figure #.** Initial Pylint report for tic\_tac\_toe module

**Appendix A**

Initial Pylint results for tic\_tac\_toe (sorted)

tic\_tac\_toe.py:163:4: C0103: Method name "game\_mode\_2\_PvE" doesn't conform to snake\_case naming style (invalid-name)

tic\_tac\_toe.py:243:4: C0103: Method name "game\_mode\_1\_PvP" doesn't conform to snake\_case naming style (invalid-name)

tic\_tac\_toe.py:1:0: C0114: Missing module docstring (missing-module-docstring)

tic\_tac\_toe.py:15:4: C0116: Missing function or method docstring (missing-function-docstring)

tic\_tac\_toe.py:21:4: C0116: Missing function or method docstring (missing-function-docstring)

tic\_tac\_toe.py:33:4: C0116: Missing function or method docstring (missing-function-docstring)

tic\_tac\_toe.py:37:4: C0116: Missing function or method docstring (missing-function-docstring)

tic\_tac\_toe.py:47:4: C0116: Missing function or method docstring (missing-function-docstring)

tic\_tac\_toe.py:52:4: C0116: Missing function or method docstring (missing-function-docstring)

tic\_tac\_toe.py:56:4: C0116: Missing function or method docstring (missing-function-docstring)

tic\_tac\_toe.py:62:4: C0116: Missing function or method docstring (missing-function-docstring)

tic\_tac\_toe.py:69:4: C0116: Missing function or method docstring (missing-function-docstring)

tic\_tac\_toe.py:79:4: C0116: Missing function or method docstring (missing-function-docstring)

tic\_tac\_toe.py:83:4: C0116: Missing function or method docstring (missing-function-docstring)

tic\_tac\_toe.py:101:4: C0116: Missing function or method docstring (missing-function-docstring)

tic\_tac\_toe.py:112:4: C0116: Missing function or method docstring (missing-function-docstring)

tic\_tac\_toe.py:120:4: C0116: Missing function or method docstring (missing-function-docstring)

tic\_tac\_toe.py:149:4: C0116: Missing function or method docstring (missing-function-docstring)

tic\_tac\_toe.py:152:4: C0116: Missing function or method docstring (missing-function-docstring)

tic\_tac\_toe.py:12:0: C0301: Line too long (113/100) (line-too-long)

tic\_tac\_toe.py:57:0: C0301: Line too long (101/100) (line-too-long)

tic\_tac\_toe.py:73:0: C0301: Line too long (102/100) (line-too-long)

tic\_tac\_toe.py:75:0: C0301: Line too long (102/100) (line-too-long)

tic\_tac\_toe.py:115:0: C0301: Line too long (111/100) (line-too-long)

tic\_tac\_toe.py:30:54: C0303: Trailing whitespace (trailing-whitespace)

tic\_tac\_toe.py:41:33: C0303: Trailing whitespace (trailing-whitespace)

tic\_tac\_toe.py:43:21: C0303: Trailing whitespace (trailing-whitespace)

tic\_tac\_toe.py:48:95: C0303: Trailing whitespace (trailing-whitespace)

tic\_tac\_toe.py:49:78: C0303: Trailing whitespace (trailing-whitespace)

tic\_tac\_toe.py:72:65: C0303: Trailing whitespace (trailing-whitespace)

tic\_tac\_toe.py:85:82: C0303: Trailing whitespace (trailing-whitespace)

tic\_tac\_toe.py:89:65: C0303: Trailing whitespace (trailing-whitespace)

tic\_tac\_toe.py:96:85: C0303: Trailing whitespace (trailing-whitespace)

tic\_tac\_toe.py:107:82: C0303: Trailing whitespace (trailing-whitespace)

tic\_tac\_toe.py:113:59: C0303: Trailing whitespace (trailing-whitespace)

tic\_tac\_toe.py:114:65: C0303: Trailing whitespace (trailing-whitespace)

tic\_tac\_toe.py:119:40: C0303: Trailing whitespace (trailing-whitespace)

tic\_tac\_toe.py:150:44: C0303: Trailing whitespace (trailing-whitespace)

tic\_tac\_toe.py:151:54: C0303: Trailing whitespace (trailing-whitespace)

tic\_tac\_toe.py:160:25: C0303: Trailing whitespace (trailing-whitespace)

tic\_tac\_toe.py:168:57: C0303: Trailing whitespace (trailing-whitespace)

tic\_tac\_toe.py:171:31: C0303: Trailing whitespace (trailing-whitespace)

tic\_tac\_toe.py:173:52: C0303: Trailing whitespace (trailing-whitespace)

tic\_tac\_toe.py:180:64: C0303: Trailing whitespace (trailing-whitespace)

tic\_tac\_toe.py:196:23: C0303: Trailing whitespace (trailing-whitespace)

tic\_tac\_toe.py:206:63: C0303: Trailing whitespace (trailing-whitespace)

tic\_tac\_toe.py:211:21: C0303: Trailing whitespace (trailing-whitespace)

tic\_tac\_toe.py:217:65: C0303: Trailing whitespace (trailing-whitespace)

tic\_tac\_toe.py:220:25: C0303: Trailing whitespace (trailing-whitespace)

tic\_tac\_toe.py:226:61: C0303: Trailing whitespace (trailing-whitespace)

tic\_tac\_toe.py:228:25: C0303: Trailing whitespace (trailing-whitespace)

tic\_tac\_toe.py:230:36: C0303: Trailing whitespace (trailing-whitespace)

tic\_tac\_toe.py:241:0: C0303: Trailing whitespace (trailing-whitespace)

tic\_tac\_toe.py:248:57: C0303: Trailing whitespace (trailing-whitespace)

tic\_tac\_toe.py:251:31: C0303: Trailing whitespace (trailing-whitespace)

tic\_tac\_toe.py:253:52: C0303: Trailing whitespace (trailing-whitespace)

tic\_tac\_toe.py:260:64: C0303: Trailing whitespace (trailing-whitespace)

tic\_tac\_toe.py:273:48: C0303: Trailing whitespace (trailing-whitespace)

tic\_tac\_toe.py:274:63: C0303: Trailing whitespace (trailing-whitespace)

tic\_tac\_toe.py:277:36: C0303: Trailing whitespace (trailing-whitespace)

tic\_tac\_toe.py:280:23: C0303: Trailing whitespace (trailing-whitespace)

tic\_tac\_toe.py:285:65: C0303: Trailing whitespace (trailing-whitespace)

tic\_tac\_toe.py:288:25: C0303: Trailing whitespace (trailing-whitespace)

tic\_tac\_toe.py:294:61: C0303: Trailing whitespace (trailing-whitespace)

tic\_tac\_toe.py:296:25: C0303: Trailing whitespace (trailing-whitespace)

tic\_tac\_toe.py:298:36: C0303: Trailing whitespace (trailing-whitespace)

tic\_tac\_toe.py:304:30: C0303: Trailing whitespace (trailing-whitespace)

tic\_tac\_toe.py:115:0: C0325: Unnecessary parens after 'if' keyword (superfluous-parens)

tic\_tac\_toe.py:127:0: C0325: Unnecessary parens after 'if' keyword (superfluous-parens)

tic\_tac\_toe.py:128:0: C0325: Unnecessary parens after 'return' keyword (superfluous-parens)

tic\_tac\_toe.py:134:0: C0325: Unnecessary parens after 'if' keyword (superfluous-parens)

tic\_tac\_toe.py:135:0: C0325: Unnecessary parens after 'return' keyword (superfluous-parens)

tic\_tac\_toe.py:140:0: C0325: Unnecessary parens after 'if' keyword (superfluous-parens)

tic\_tac\_toe.py:141:0: C0325: Unnecessary parens after 'return' keyword (superfluous-parens)

tic\_tac\_toe.py:146:0: C0325: Unnecessary parens after 'if' keyword (superfluous-parens)

tic\_tac\_toe.py:147:0: C0325: Unnecessary parens after 'return' keyword (superfluous-parens)

tic\_tac\_toe.py:2:0: C0411: standard import "import random" should be placed before "import numpy as np" (wrong-import-order)

tic\_tac\_toe.py:3:0: C0411: standard import "from time import sleep" should be placed before "import numpy as np" (wrong-import-order)

tic\_tac\_toe.py:120:4: R0912: Too many branches (14/12) (too-many-branches)

tic\_tac\_toe.py:41:8: R1705: Unnecessary "else" after "return", remove the "else" and de-indent the code inside it (no-else-return)

tic\_tac\_toe.py:155:8: R1705: Unnecessary "elif" after "return", remove the leading "el" from "elif" (no-else-return)

tic\_tac\_toe.py:120:4: R1710: Either all return statements in a function should return an expression, or none of them should. (inconsistent-return-statements)

tic\_tac\_toe.py:152:4: R1710: Either all return statements in a function should return an expression, or none of them should. (inconsistent-return-statements)

tic\_tac\_toe.py:39:15: R1714: Consider merging these comparisons with "in" to "marker not in ('X', 'O')" (consider-using-in)

tic\_tac\_toe.py:90:16: R1722: Consider using sys.exit() (consider-using-sys-exit)

tic\_tac\_toe.py:91:12: W0702: No exception type(s) specified (bare-except)

tic\_tac\_toe.py:169:29: W1309: Using an f-string that does not have any interpolated variables (f-string-without-interpolation)

tic\_tac\_toe.py:249:29: W1309: Using an f-string that does not have any interpolated variables (f-string-without-interpolation)

tic\_tac\_toe.py:95:21: W1514: Using open without explicitly specifying an encoding (unspecified-encoding)

tic\_tac\_toe.py:107:13: W1514: Using open without explicitly specifying an encoding (unspecified-encoding)

tic\_tac\_toe.py:168:17: W1514: Using open without explicitly specifying an encoding (unspecified-encoding)

tic\_tac\_toe.py:217:25: W1514: Using open without explicitly specifying an encoding (unspecified-encoding)

tic\_tac\_toe.py:225:25: W1514: Using open without explicitly specifying an encoding (unspecified-encoding)

tic\_tac\_toe.py:248:17: W1514: Using open without explicitly specifying an encoding (unspecified-encoding)

tic\_tac\_toe.py:285:25: W1514: Using open without explicitly specifying an encoding (unspecified-encoding)

tic\_tac\_toe.py:293:25: W1514: Using open without explicitly specifying an encoding (unspecified-encoding)

Graphical user interface

Description automatically generated

**Figure #.** Initial Pylint report for test module