

Pynguin: Tool Analysis

Team members: Iliescu Andrei, Iancu Aurelian, Gligor Ovidiu

Getting started:

To get started simply use pip:

```
pip install pynguin
```

This is the class that we are going to test using pynguin:

```
class Queue:
    def __init__(self, size_max: int) -> None:
        assert size_max > 0
        self.max = size_max
        self.head = 0
        self.tail = 0
        self.size = 0
        self.data = array.array("i", range(size_max))

    def empty(self) -> bool:
        return self.size != 0

    def full(self) -> bool:
        return self.size == self.max

    def enqueue(self, x: int) -> bool:
        if self.size == self.max:
            return False
        self.data[self.tail] = x
        self.size += 1
        self.tail += 1
        if self.tail == self.max:
            self.tail = 0
        return True

    def dequeue(self) -> int | None:
        if self.size == 0:
            return None
        x = self.data[self.head]
        self.size -= 1
        self.head += 1
        if self.head == self.max:
            self.head = 0
        return x
```

After that you should be able to run a command like this:

```
pynguin \  
  --project-path ./docs/source/_static \  
  --output-path /tmp/pynguin-results \  
  --module-name example \  
  -v  
python -m penguyn
```

And everything should work but unfortunately I runned into some problems that you might run into as well and here is how I fixed them:

- First thing to keep in mind is that I used vscode for the entire time
- After installing pynguin with pip I tried simply calling it like above but it didn't work
- After some research I found that you should use python 3.10 and I was using python 3.11
- After installing python 3.10 I had to choose this version as the python interpreter in vscode. Then I had to restart vscode.
- Then again I tried simply running pynguin and again no luck so the fix was running pynguin with python as a module like this: `python -m penguyn [other-arguments]`
- I again got an error saying that environment variable PYNGUIN_DANGER_AWARE is not set. So the final fix was to go into the system environment variables table, add a new environment variable with the name PYNGUIN_DANGER_AWARE and the value 1, then restart vscode and run the command one more time and boom!...it finally worked!

```
python -m pynguin --project_path "D:\IntelliJ Projects\Pynguin" --output_path "D:\IntelliJ Projects\Pynguin\pynguin-result" --module_name queue_example -v  
[18:27:35] INFO Start Pynguin Test Generation.  
INFO Collecting static constants from module under test  
INFO No constants found  
INFO Setting up runtime collection of constants  
INFO Analyzed project to create test cluster  
INFO Modules: 2  
INFO Functions: 0  
INFO Classes: 13  
INFO Using seed 1713281253569713700  
INFO Using strategy: Algorithm.DYNAMOSA  
INFO Instantiated 14 fitness functions  
INFO Using CoverageArchive  
INFO Using selection function: Selection.TOURNAMENT_SELECTION  
INFO No stopping condition configured!  
INFO Using fallback timeout of 600 seconds  
INFO Using crossover function: SinglePointRelativeCrossOver  
INFO Using ranking function: RankBasedPreferenceSorting  
INFO Start generating test cases  
generator.py:187  
generator.py:207  
generator.py:218  
generator.py:219  
module.py:1344  
module.py:1345  
module.py:1346  
module.py:1347  
generator.py:193  
generationalgorithmfactory.py:302  
generationalgorithmfactory.py:393  
generationalgorithmfactory.py:346  
generationalgorithmfactory.py:321  
generationalgorithmfactory.py:119  
generationalgorithmfactory.py:120  
generationalgorithmfactory.py:334  
generationalgorithmfactory.py:354  
generator.py:517
```

```

[18:27:38] INFO      Running tests on mutant 17/31 assertiongenerator.py:295
INFO      Running tests on mutant 18/31 assertiongenerator.py:295
INFO      Running tests on mutant 19/31 assertiongenerator.py:295
INFO      Running tests on mutant 20/31 assertiongenerator.py:295
INFO      Running tests on mutant 21/31 assertiongenerator.py:295
INFO      Running tests on mutant 22/31 assertiongenerator.py:295
INFO      Running tests on mutant 23/31 assertiongenerator.py:295
INFO      Running tests on mutant 24/31 assertiongenerator.py:295
INFO      Running tests on mutant 25/31 assertiongenerator.py:295
INFO      Running tests on mutant 26/31 assertiongenerator.py:295
INFO      Running tests on mutant 27/31 assertiongenerator.py:295
INFO      Running tests on mutant 28/31 assertiongenerator.py:295
INFO      Running tests on mutant 29/31 assertiongenerator.py:295
INFO      Running tests on mutant 30/31 assertiongenerator.py:295
INFO      Running tests on mutant 31/31 assertiongenerator.py:295
INFO      Mutant 0 killed by Test(s): 1, 2, 3, 4, 5, 6 assertiongenerator.py:374
INFO      Mutant 1 killed by Test(s): 1, 3, 4, 5, 6 assertiongenerator.py:374
INFO      Mutant 2 killed by Test(s): 1, 3, 5, 6 assertiongenerator.py:374
INFO      Mutant 3 killed by Test(s): 1, 3, 5, 6 assertiongenerator.py:374
INFO      Mutant 4 killed by Test(s): 1, 2, 3, 4, 5, 6 assertiongenerator.py:374
INFO      Mutant 5 killed by Test(s): 1, 3, 4, 5, 6 assertiongenerator.py:374
INFO      Mutant 6 killed by Test(s): 1, 3, 5, 6 assertiongenerator.py:374
INFO      Mutant 7 killed by Test(s): 1, 3, 5, 6 assertiongenerator.py:374
INFO      Mutant 8 killed by Test(s): 0, 1, 2, 5, 6 assertiongenerator.py:374
INFO      Mutant 9 killed by Test(s): 0, 1, 2, 3, 4, 5, 6 assertiongenerator.py:374
INFO      Mutant 10 killed by Test(s): 0, 1, 2, 3, 4, 5, 6 assertiongenerator.py:374
INFO      Mutant 11 killed by Test(s): 0, 1, 2, 3, 4, 5, 6 assertiongenerator.py:374
INFO      Mutant 12 killed by Test(s): 0, 1, 2, 3, 4, 5, 6 assertiongenerator.py:374
INFO      Mutant 13 killed by Test(s): 0, 1, 2, 3, 4, 5, 6 assertiongenerator.py:374
INFO      Mutant 14 killed by Test(s): 1, 4, 5, 6 assertiongenerator.py:374
INFO      Mutant 15 killed by Test(s): 1, 2, 3, 4, 5, 6 assertiongenerator.py:374
INFO      Mutant 16 killed by Test(s): 1, 3, 4, 5, 6 assertiongenerator.py:374
INFO      Mutant 17 killed by Test(s): 1, 5, 6 assertiongenerator.py:374
INFO      Mutant 18 killed by Test(s): 1, 3, 5, 6 assertiongenerator.py:374
INFO      Mutant 19 killed by Test(s): 1, 3, 5, 6 assertiongenerator.py:374
INFO      Mutant 20 killed by Test(s): 1, 3, 5, 6 assertiongenerator.py:374
INFO      Mutant 21 killed by Test(s): 1, 5, 6 assertiongenerator.py:374
INFO      Mutant 22 killed by Test(s): 1, 5, 6 assertiongenerator.py:374
INFO      Mutant 23 killed by Test(s): 0, 1, 2, 3, 4, 5, 6 assertiongenerator.py:374
INFO      Mutant 24 killed by Test(s): 1, 5, 6 assertiongenerator.py:374
INFO      Mutant 25 killed by Test(s): 1, 4, 5, 6 assertiongenerator.py:374
INFO      Mutant 26 killed by Test(s): 1, 3, 5, 6 assertiongenerator.py:374
INFO      Mutant 27 killed by Test(s): 1, 2, 3, 4, 5, 6 assertiongenerator.py:374
INFO      Mutant 28 killed by Test(s): 1, 3, 4, 5, 6 assertiongenerator.py:374
INFO      Mutant 29 killed by Test(s): 1, 3, 5, 6 assertiongenerator.py:374
INFO      Mutant 30 killed by Test(s): 1, 3, 5, 6 assertiongenerator.py:374
INFO      Number of Surviving Mutant(s): 0 (Mutants: ) assertiongenerator.py:386
INFO      Calculating resulting FinalBranchCoverage generator.py:438
INFO      Written 7 test cases to D:\IntelliJ Projects\Pynguin\pynguin-result\test_queue_example.py generator.py:788
INFO      Writing statistics statistics.py:369
INFO      Stop Pynguin Test Generation... generator.py:110

```

And the file that was generated:

```

3 import pytest
4 import queue_example as module_0
5
6
7 def test_case_0():
8     bool_0 = True
9     queue_0 = module_0.Queue(bool_0)
10    assert (
11        f"{type(queue_0).__module__}.{type(queue_0).__qualname__}"
12        == "queue_example.Queue"
13    )
14    assert queue_0.max is True
15    assert queue_0.head == 0
16    assert queue_0.tail == 0
17    assert queue_0.size == 0
18    assert (
19        f"{type(queue_0.data).__module__}.{type(queue_0.data).__qualname__}"
20        == "array.array"
21    )
22    assert len(queue_0.data) == 1
23    var_0 = queue_0.dequeue()
24
25
26 def test_case_1():
27     bool_0 = False
28     with pytest.raises(AssertionError):
29         module_0.Queue(bool_0)
30
31
32 def test_case_2():
33     int_0 = -2314
34     bool_0 = True
35     queue_0 = module_0.Queue(bool_0)
36    assert (

```