

TOOL DEMO

Penguin

DIANY PRESSATO

# WHAT IS PYNGUIN?

- Automated **unit test** generation tool for **Python**
- Search-Based (genetic algorithm)
- Increases code **coverage**





For Python!

ICSE'22

WHY LIKE



1.3K stars on  
GitHub





Well maintained!

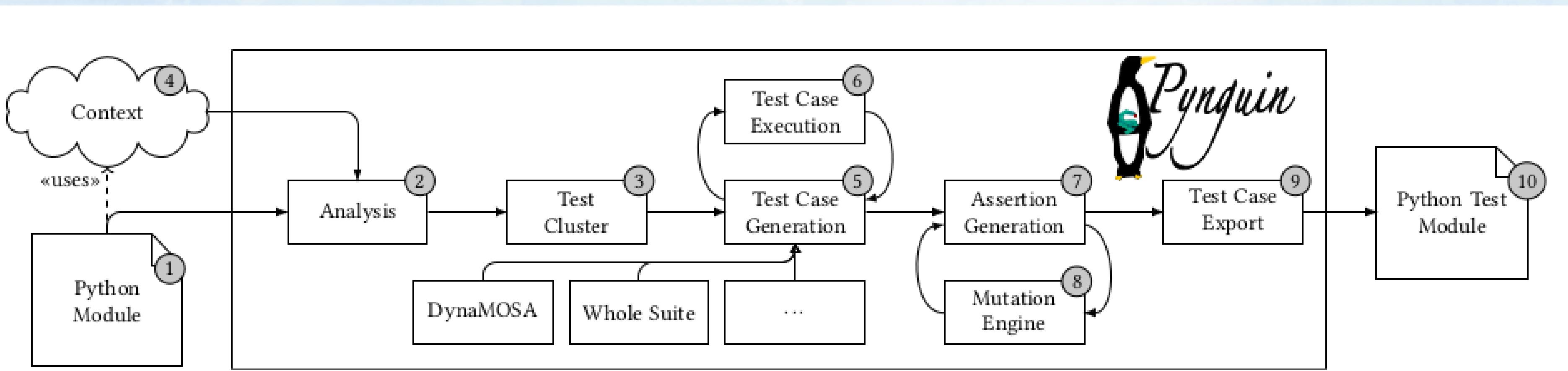
Dynamically  
typed!

Automatic!

WHY LIKE



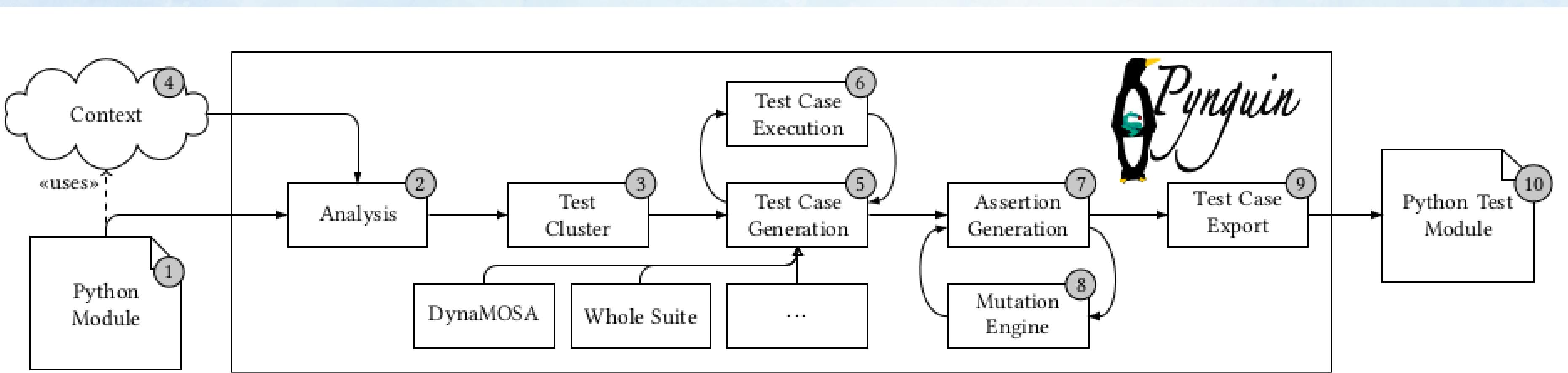
# EXECUTION STEPS



## Context:

- extract types from class definitions available in the module namespace

# EXECUTION STEPS



# Test Cluster:

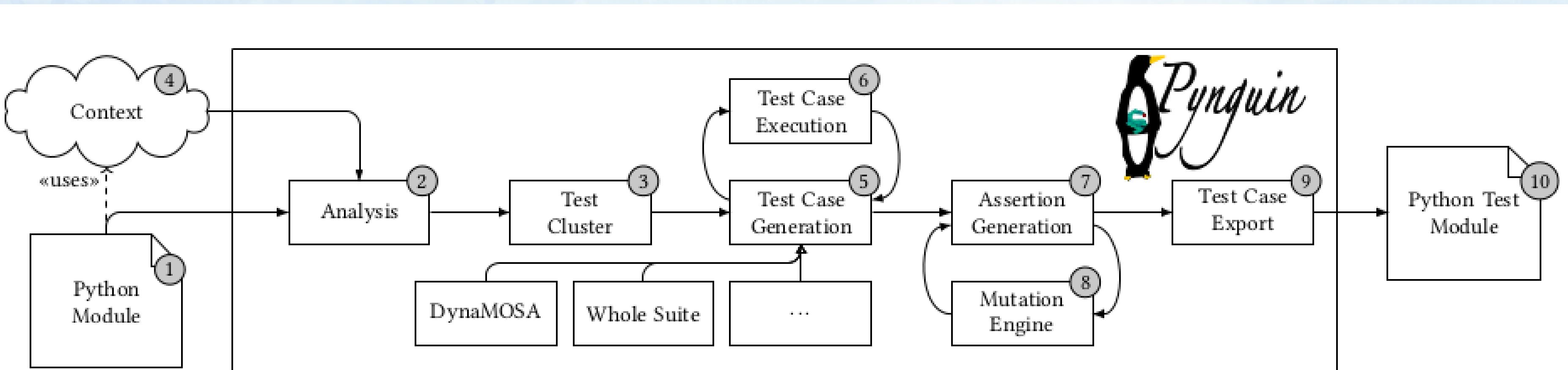
## module under test

## methods, functions, classes

## parameters

- Three dark blue circular dots are arranged vertically along the right edge of the page.

# EXECUTION STEPS

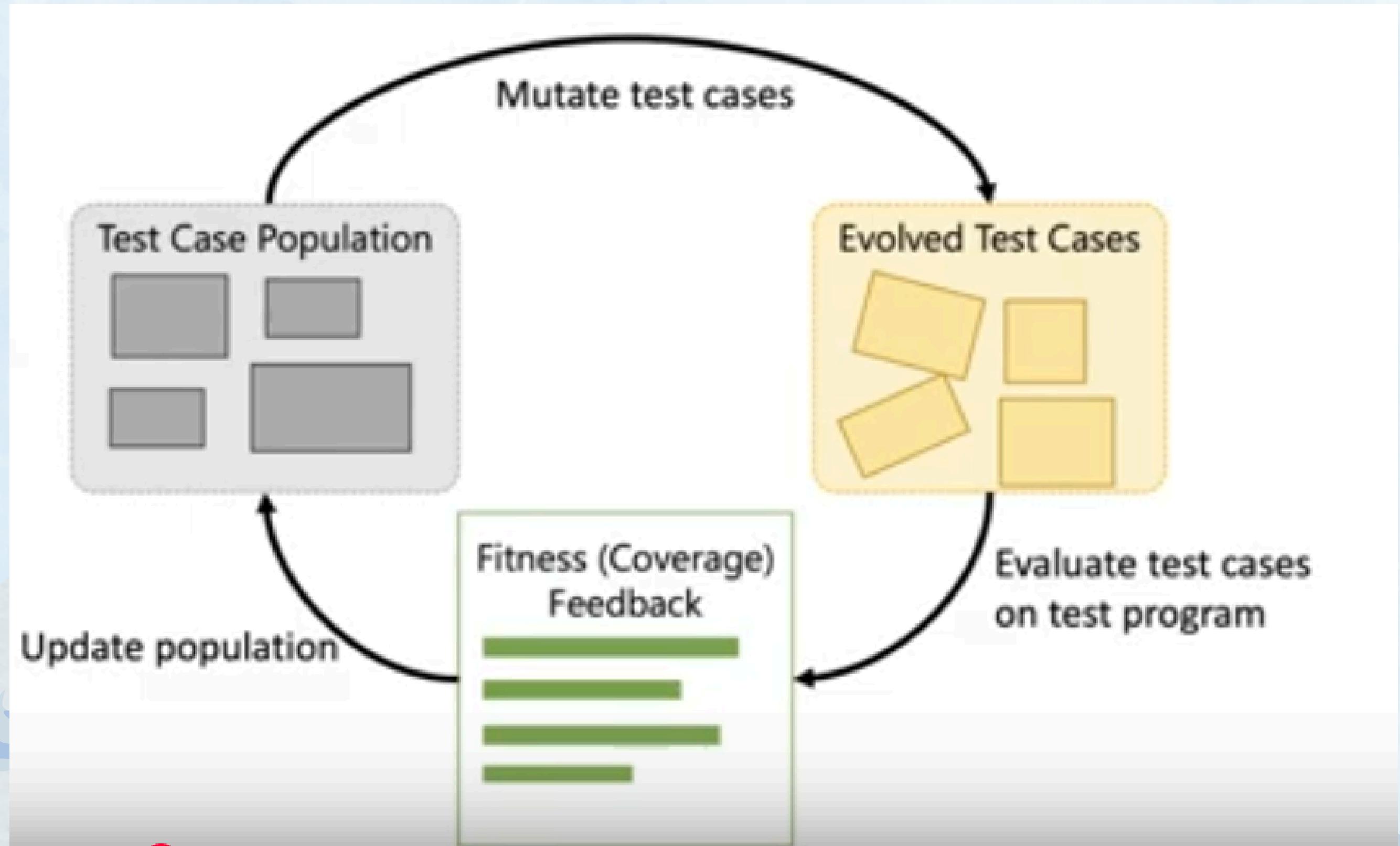


# Algorithms

# DynaMOSA MOSA

- # MIO Feedback-Directed Random Whole Suit

# GENETIC ALGORITHMS



# DYNAMICALLY TYPED

```
x = 42          # x is an integer  
x = "hello"    # Now x is a string  
x = [1, 2, 3]  # Now x is a list
```

variable's type can change throughout execution!

# DYNAMICALLY-TYPED

```
def process(data):
    if isinstance(data, int):
        return data * 2
    elif isinstance(data, str):
        return data.upper()
    else:
        return None
```

Distinct return types!

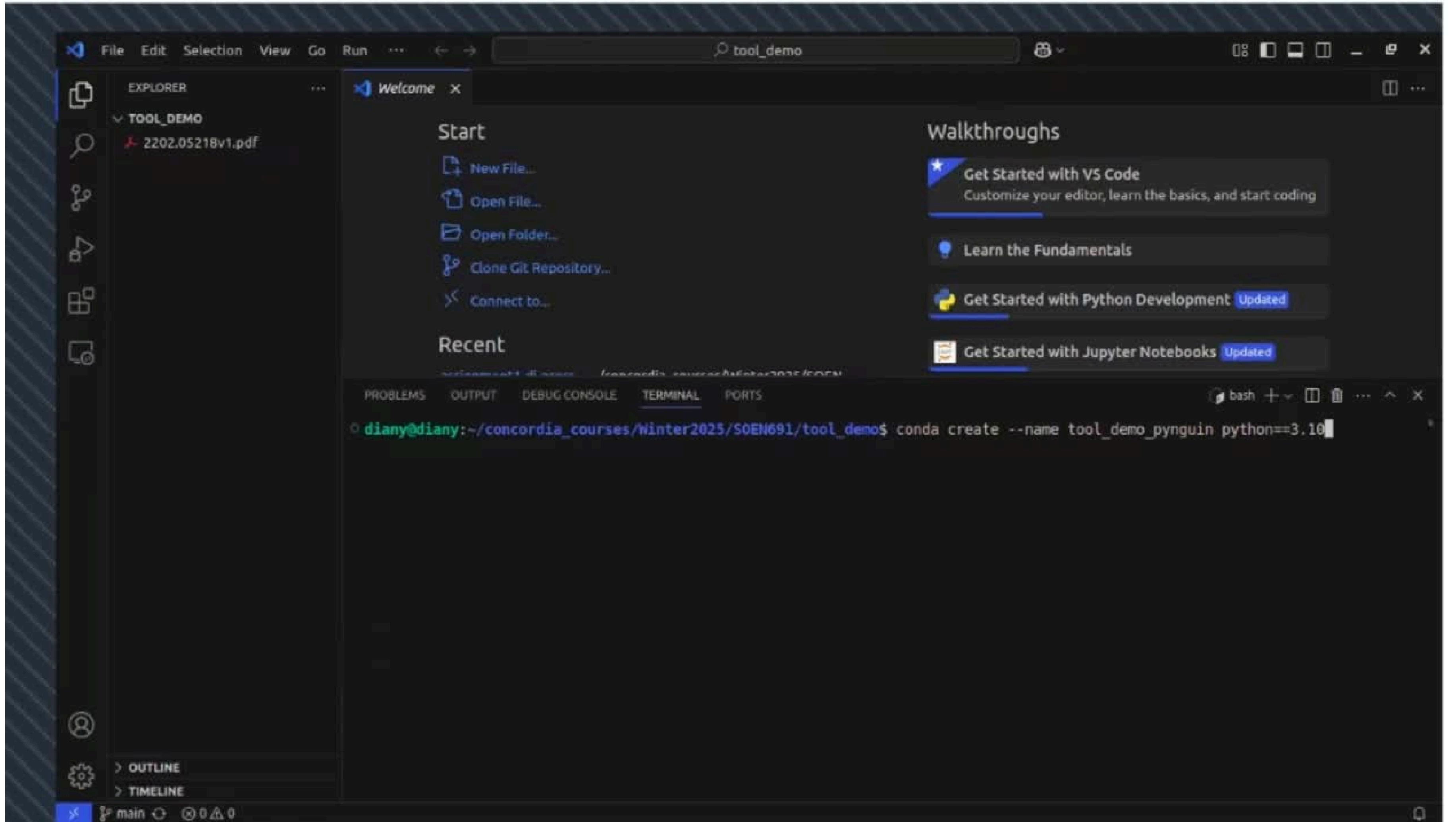
# RESEARCH QUESTION

How the different algorithms employed by Pynguin tool affect the overall performance of code / unit test generation done by LLMs?



**DEMO**





# REFERENCES

Stephan Lukasczyk and Gordon Fraser. 2022. Pynguin: automated unit test generation for Python. In Proceedings of the ACM/IEEE 44th International Conference on Software Engineering: Companion Proceedings (ICSE '22). Association for Computing Machinery, New York, NY, USA, 168–172.  
<https://doi.org/10.1145/3510454.3516829>

<https://dl.acm.org/doi/10.1145/3510454.3516829>

<https://github.com/se2p/pynguin>

<https://www.youtube.com/watch?v=DOuKlfNtT7M&t=392s>