1. Lexical analysis:

- Comments and unneeded spaces are removed.
- Keywords are identified

Example:

```
#Using maths in Python

print("Calculator program")

prant("Adding two numbers")

first = input("Enter a number")

answer = first +100

print(answer)
```

print identified as a keyword

```
print("Calculator program")
prant("Adding two numbers")
first=input("Enter a number")
answer= first +100
print(answer)
```

2. Syntax analysis

- Spelling and grammar checked to match the particular language
- If syntax errors are found, error messages are produced.

Example:

NameError: name 'prant' is not defined

```
print("Calculator program")
prant("Adding two numbers")
first=input("Enter a number")
answer= first +100
print(answer)
```

3. Semantic analysis

- Variables are checked to ensure they have the correct data type and being used correctly

Variable 'first' should be an integer data type

```
answer= first +100
TypeError: must be str, not int
```

```
prant("Adding two numbers")
first=input("Enter a number")
answer= first +100
```

4. Code generation

- Machine code is generated. Example:

```
10100010 1010010100 101010010 1
10001001 10101010110 101010101
```

5. Code optimisation

- Variables are checked to ensure that they have been properly declared and used.

Keyword – *Optimisation: the action of making the best or most effective use of a situation or resource.*