Grammar for fractals

Team 16: Botnari Ciprian, Guzun Grigore,
Cucoș Emanuil, Mirovskki Artiom

Presentation

01 **Goal**
The ultimate objective of DSL

02 **Grammar**
The structure of the language

03 **Program**
A sneak peek of how to create a fractal

04 **Fractal**
What we strive for to achieve

05 **Language**
Future of our DSL

06 **Conclusions**
Final acknowledgements

Goal

- Fractals are complex mathematical objects that can be used to teach various mathematical and computer science concepts.
- However, building them can be a challenging task, especially for educators and students who are not familiar with complex programming languages.
- The goal of this DSL is to provide an easy-to-use and intuitive language for educators and students to build and explore fractals without the need for extensive programming knowledge.

Grammar (Work in Progress)

G = (V_N, V_T, P, S)

V_N = {REPEAT, TIMES, START, WITH, SHAPE, CIRCLE, SQUARE, TRIANGLE, COLOR, BACKGROUND, SCALE, ROTATE, SAVE, AS, PNG, JPG, [A-Z], [a-z], [0-9], =, ., ,, [,]}

V_T = { <start>, <command>, <shape>, <transformation>, <color>, <repeat>, <save>, <lowercase>, <uppercase>, <number>}

P = { <start> -> command
<command> -> shape | transformation | color | repeat | save
<shape> -> circle(value) | square(value) | triangle(value) | polygon(sides, value)
<transformation> -> scale(value) | rotate(value)
<color> -> red | blue | green | yellow | black | white
<repeat> -> repeat(value) { command }
<save> -> save as file_type(filename)
<lowercase→ a – z; <uppercase> → A – Z; <number> → 0 - 9}
S = <start>

Program

```
// This is a comment

// Define the initial shape and settings
start with shape circle(100);
color blue;
background white;

// Apply transformations
repeat 5 times {
  scale(0.5);
  rotate(60);
  color red;
  shape square(50);
}

// Apply more transformations
repeat 10 times {
  scale(0.8);
  rotate(30);
  color green;
  shape triangle(50);
}

// Save the fractal as an image
save as PNG("my_fractal.png");
```

Fractal...

Next steps

Lexical Analyzer

A software component that reads and analyzes the characters of a program's source code and generates a sequence of tokens that the compiler or interpreter can use

Symbol Table

A data structure that stores information about the identifiers used in a program, such as their name, type, and scope.

Scientific paper

In our scientific paper, we propose a domain-specific language (DSL) for building fractals that is specifically designed for educators and students.

Thank you for the attention