

## **Компилятор языка C# на языке Python**

**Ушаков Владимир, Агафонова Марина, Семечев Данила(все – 4.2)**

### **Компилятор на языке Python с помощью библиотеки `ruparsing`: AST-дерево**

При построении AST-дерева реализовано:

`return`

Инициализация массива (с заданием элементов или через `new <type>` и указанием размеров)

Множественное присваивание значений переменным

Математические и логические операции

Вызов функций

Тип `string`

Циклы (`for`, `while`)

Условный оператор (`if`)

Объявление функций

Объявление простейших классов

Выбор поля у класса через `.`

Выбор значения через `[]` (пока только для одномерного массива)

Объявление объектов класса через `new <type>`

### **Структура проекта**

Проект состоит из:

- файлов грамматики, в которых расписана вся грамматика парсинга входящего кода
- файлов классов узлов построения AST-дерева
- `main`

## Основные классы дерева:

- AstNode – абстрактный класс узла дерева
- ExprNode – выражение
- IdentNode – тип объекта
- LiteralNode – символ
- BinOpNode – узел бинарных операций
- StmtNode – узел кода
- CallNode – узел вызова функций
- ArrayNode/ClazzDecNode/FuncDecNode – объявление массива/ класса/ функции
- ArrayNewInitNode/ClazzNewInitNode – идентификация массива/ класса через new
- AssignNode – присваивание
- WhileNode/ ForNode – циклы while/ for
- IfNode – условие if
- ReturnNode – оператор возврата return

## Примеры вход-выход

Вход\_1:

```
class A
```

```
{  
    int a(int a)  
    {  
        int b = 0;  
    }  
  
    int b;  
}
```

```
int a(int a)  
{  
    int b = 0;
```

}

A a1 = new A();

Выход\_1:

...

| class\_decl

| | A

| | func\_dec

| | | int

| | | a

| | | arg

| | | | int

| | | | L a

| | | L ...

| | | L var

| | | | int

| | | | L =

| | | | | b

| | | | | L 0.0 (float)

| | L var

| | | int

| | L b

| func\_dec

| | int

| | a

| | arg

| | | int

| | | L a

| | L ...

```

|   L var
|   └ int
|   L =
|   └ b
|   L 0.0 (float)
L ...
L var
  └ A
    L =
      └ a1
        L new class
          L A

```

Вход\_2:

```

for (int i = 0, j = 8; ((i <= 5)) && g; i = i + 1, print(5))
    for(; a < b;)
        if (a > 7 + b) {
            c = a+b * (2 - 1) + 0; // comment 2
            b = "98\typa";
        }
        else if (f)
            output(c + 1, 89.89);

while (g2 > g)
    output(g2, g5);

```

ВЫХОД\_2:

...

└ ...

└ for

└ └ var

└ └ └ int

└ └ └ =

└ └ └ └ i

└ └ └ └ └ 0.0 (float)

└ └ └ =

└ └ └ j

└ └ └ └ 8.0 (float)

└ └ └ &&

└ └ └ <=

└ └ └ └ i

└ └ └ └ └ 5.0 (float)

└ └ └ g

└ └ └ ...

└ └ └ =

└ └ └ └ i

└ └ └ └ └ +

└ └ └ └ └ i

└ └ └ └ └ └ 1.0 (float)

└ └ └ └ call

└ └ └ └ print

└ └ └ └ └ 5.0 (float)

└ └ └ for

└ └ └ ...

└ └ └ <

```

| | ⊢ a
| | ⊢ b
| ⊢ ...
| ⊢ if
|   ⊢ >
|   | ⊢ a
|   | ⊢ +
|   | ⊢ 7.0 (float)
|   | ⊢ b
|   ⊢ ...
|   | ⊢ =
|   | | ⊢ c
|   | | ⊢ +
|   | | ⊢ +
|   | | | ⊢ a
|   | | | ⊢ *
|   | | | ⊢ b
|   | | | ⊢ -
|   | | | ⊢ 2.0 (float)
|   | | | ⊢ 1.0 (float)
|   | | ⊢ 0.0 (float)
|   | ⊢ =
|   | ⊢ b
|   | ⊢ "98 ypa" (str)
|   ⊢ if
|     ⊢ f
|     ⊢ call
|     ⊢ output
|     ⊢ +

```

```

|      | ⊢ c
|      | ⊢ 1.0 (float)
|      | ⊢ 89.89 (float)
⊢ while
  ⊢ >
  | ⊢ g2
  | ⊢ g
  ⊢ call
    ⊢ output
    ⊢ g2
    ⊢ g5

```

Вход\_3:

```

int[] b = { a, b , {c, d } };
    a = a.lenght;
    a.lenght();
    //a = a[b + c];
    return ;
    return 2;
    return a+b;
    a = a[2];
    int[] b = new int[b + c][4];
    int[] b = 4;
    int g, g2 = g, g = 90;

    a = input(); b = input(); /* comment 1
    c = input();
    */

    //int [ ] b = int [ 5 ];

```

```
//a = int[5];
```

```
string b = "Привет";
```

Выход\_3:

...

└ ...

└ var

└ └ □

└ └ └  $\vdash$  int

└ └  $\vdash$  =

└ └ └ b

└ └ └  $\vdash$  { }

└ └ └ a

└ └ └ b

└ └ └  $\vdash$  { }

└ └ └ c

└ └ └  $\vdash$  d

└  $\vdash$  =

└ └ a

└ └  $\vdash$  .

└ └ └ a

└ └ └  $\vdash$  lenght

└ └  $\vdash$  .

└ └ └ a

└ └ └  $\vdash$  call

└ └ └  $\vdash$  lenght

└ └  $\vdash$  return

└ └ └  $\vdash$  ...

└ └  $\vdash$  return

└ └ └  $\vdash$  2.0 (float)



```

└ return
|  L +
|  └ a
|  L b
└ =
|  └ a
|  L [] val
|  └ a
|  L 2.0 (float)
└ var
|  └ []
|  |  L int
|  L =
|  └ b
|  L new
|  └ int
|  └ +
|  |  └ b
|  |  L c
|  L 4.0 (float)
└ var
|  └ []
|  |  L int
|  L =
|  └ b
|  L 4.0 (float)
└ var
|  └ int
|  └ g

```

|  $\vdash =$

| |  $\vdash g2$

| |  $\vdash g$

|  $\vdash =$

|  $\vdash g$

|  $\vdash 90.0$  (float)

$\vdash =$

|  $\vdash a$

|  $\vdash \text{call}$

|  $\vdash \text{input}$

$\vdash =$

|  $\vdash b$

|  $\vdash \text{call}$

|  $\vdash \text{input}$

$\vdash \text{var}$

$\vdash \text{string}$

$\vdash =$

$\vdash b$

$\vdash \text{"Привет"}$  (str)

## Диаграмма классов

