Delimiters

⟨*singleQuote*⟩ ::= ’

⟨*doubleQuote*⟩ ::= ”

⟨*terminator*⟩ ::= ‘`;`’ | ‘`\n`’

Primitive Types

⟨*int*⟩ ::= [integer]

⟨*bool*⟩ ::= ‘`true`’ | ‘`false`’

⟨*char*⟩ ::= ⟨*singleQuote*⟩[character]⟨*singleQuote*⟩

⟨*string*⟩ ::= ⟨*doubleQuote*⟩[character*]⟨*doubleQuote*⟩

⟨*null*⟩ ::= ‘`null`’

Algebraic Data Types and Type-Traits

⟨*adt*⟩ ::= ‘`type`’ ⟨*ident*⟩ ‘{’[⟨*ident*⟩‘`:`’⟨*type*⟩[‘`,`’ ⟨*ident*⟩‘`:`’⟨*type*⟩]*]‘}’

⟨*trait*⟩ ::= ‘`trait`’ ‘{’⟨*type*⟩[‘`,`’ ⟨*type*⟩]*‘}’

Types

⟨*type*⟩ ::= ‘`int`’ | ‘`bool`’ | ‘`char`’ | ‘`string`’ | ‘`null`’
| ⟨*type*⟩ ‘`->`’ ⟨*type*⟩
| ‘`(`’[⟨*type*⟩[‘`,`’ ⟨*type*⟩]*]‘`)`’ ‘`->`’ ‘`(`’[⟨*type*⟩[‘`,`’ ⟨*type*⟩]*]‘`)`’
| ‘`List`’ | ‘`Array`’ ‘`[`’⟨*type*⟩‘`]`’
| ‘`Tuple`’ ‘`[`’⟨*type*⟩[‘`,`’ ⟨*type*⟩]*‘`]`’
| ‘`Dict`’ ‘`[`’⟨*type*⟩‘`,`’ ⟨*type*⟩‘`]`’
| ⟨*ident*⟩

Arithmetic and Boolean Operators

⟨*arithOp*⟩ ::= ‘`+`’ | ‘`-`’ | ‘`*`’ | ‘`/`’ | ‘`%`’

⟨*boolOp*⟩ ::= ‘`<`’ | ‘`>`’ | ‘`<=`’ | ‘`>=`’ | ‘`!`’ | ‘`!=`’ | ‘`==`’ | ‘`&&`’ | ‘`||`’

⟨*op*⟩ ::= ⟨*arithOp*⟩ | ⟨*boolOp*⟩

Functions

```
⟨arg⟩ ::= ⟨ident⟩‘:’⟨type⟩[‘=’⟨atom⟩]

⟨templateTypes⟩ ::= ‘[’⟨type⟩‘:>’⟨ident⟩[‘,’ ⟨type⟩‘:>’⟨ident⟩]*‘]’

⟨funDef⟩ ::= ‘fn’ ⟨ident⟩[⟨templateTypes⟩]‘(’[⟨arg⟩ [‘,’ ⟨arg⟩]* ]‘)’[‘->’⟨type⟩]‘=’⟨smp⟩⟨terminator⟩

⟨prog⟩ ::= [⟨funDef⟩]*

⟨app⟩ ::= ⟨atom⟩[ [‘[’⟨type⟩[‘,’ ⟨type⟩]*‘]’ ]* [‘(’[⟨smp⟩[‘,’ ⟨smp⟩]*]‘)’]* ]
```

Pattern Matching and Switches

```
⟨match⟩ ::= ‘match’ ‘(’⟨ident⟩‘)’ ‘{’‘case’ ⟨type⟩‘=>’ ⟨smp⟩[‘,’ ‘case’ ⟨type⟩ ‘=>’ ⟨smp⟩]*‘}’

⟨switch⟩ ::= ‘switch’ ‘(’⟨atom⟩‘)’ ‘{’‘case’ ⟨atom⟩‘=>’ ⟨smp⟩[‘,’ ‘case’ ⟨atom⟩ ‘=>’ ⟨smp⟩]*‘}’

⟨matchSwitch⟩ ::= ⟨match⟩ | ⟨switch⟩
```

Expressions

```
⟨atom⟩ ::= ⟨int⟩ | ⟨bool⟩ | ⟨char⟩ | ⟨string⟩ | ⟨null⟩
  |  ‘(’⟨smp⟩‘)’
  |  ⟨ident⟩[‘.’⟨ident⟩]*

⟨tight⟩ ::= ⟨app⟩[‘:>’⟨app⟩]
  |  [‘(’⟨smp⟩‘)’]+
  |  ’{’⟨exp⟩‘}’

⟨utight⟩ ::= [⟨op⟩]⟨tight⟩

⟨smp⟩ ::= ⟨utight⟩[⟨op⟩⟨utight⟩]
  |  ‘if’ ‘(’⟨smp⟩‘)’ ⟨smp⟩ [‘else’ ⟨smp⟩]
  |  ‘List’ | ‘Tuple’ | ‘Array’ ‘{’[⟨smp⟩[‘,’ ⟨smp⟩]*]‘}’
  |  ‘Dict’ ‘{’[⟨smp⟩‘:’⟨smp⟩[‘,’ ⟨smp⟩‘:’⟨smp⟩]*]‘}’
  |  ⟨matchSwitch⟩
  |  ⟨trait⟩
  |  ⟨adt⟩
  |  ⟨prog⟩
  |  ⟨arg⟩ ‘=>’ ⟨smp⟩

⟨exp⟩ ::= ⟨smp⟩[⟨terminator⟩⟨exp⟩]
  |  ‘val’ ⟨ident⟩[‘:’⟨type⟩] ‘=’ ⟨smp⟩⟨terminator⟩⟨exp⟩
  |  ‘include’ ⟨file⟩⟨terminator⟩⟨exp⟩
```