Functions and comma expressions

Dolphin/Phase 4
Compilation 2024

Aslan Askarov

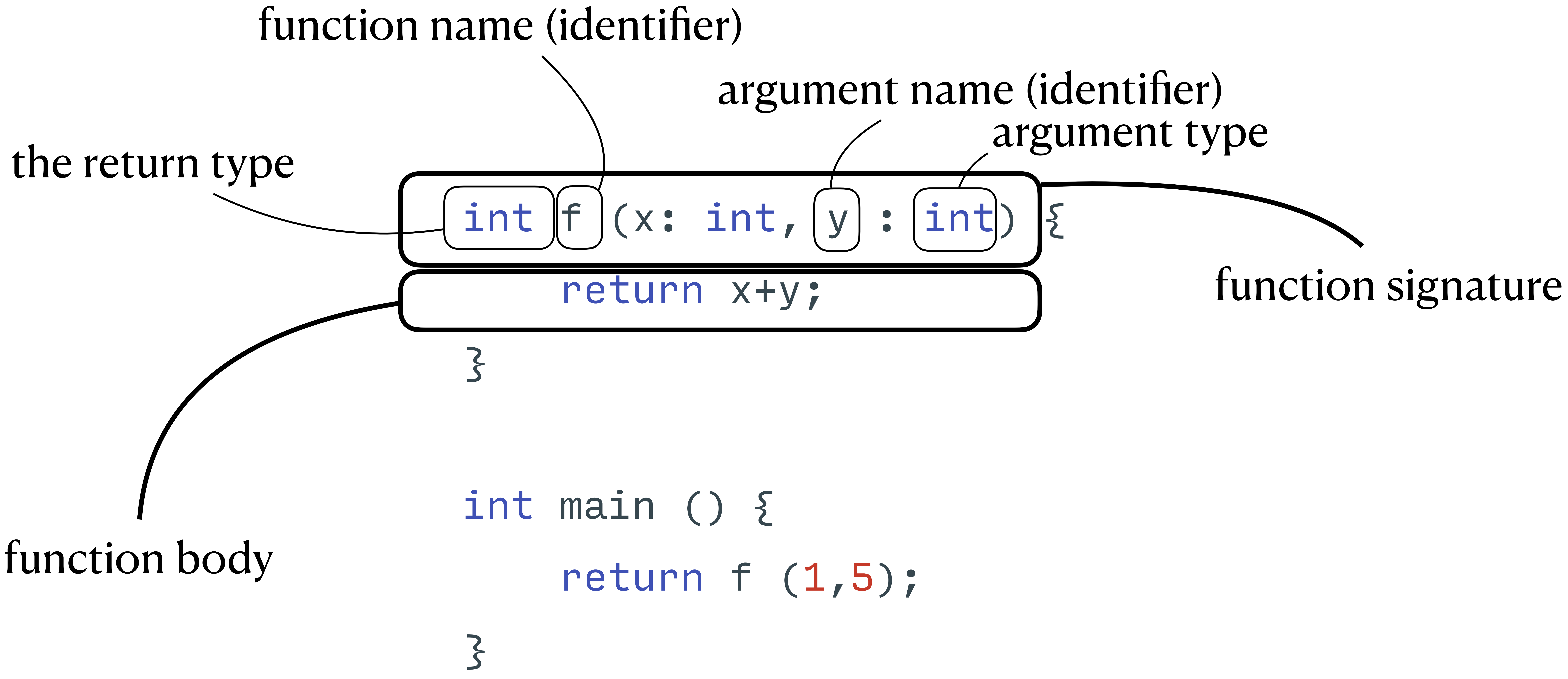
What's new in Phase 4?

- Functions:
  - C-like, top-level, mutually recursive functions
- Comma expressions:
  - C/Java-inspired sequencing of expressions; typically useful in for-loops

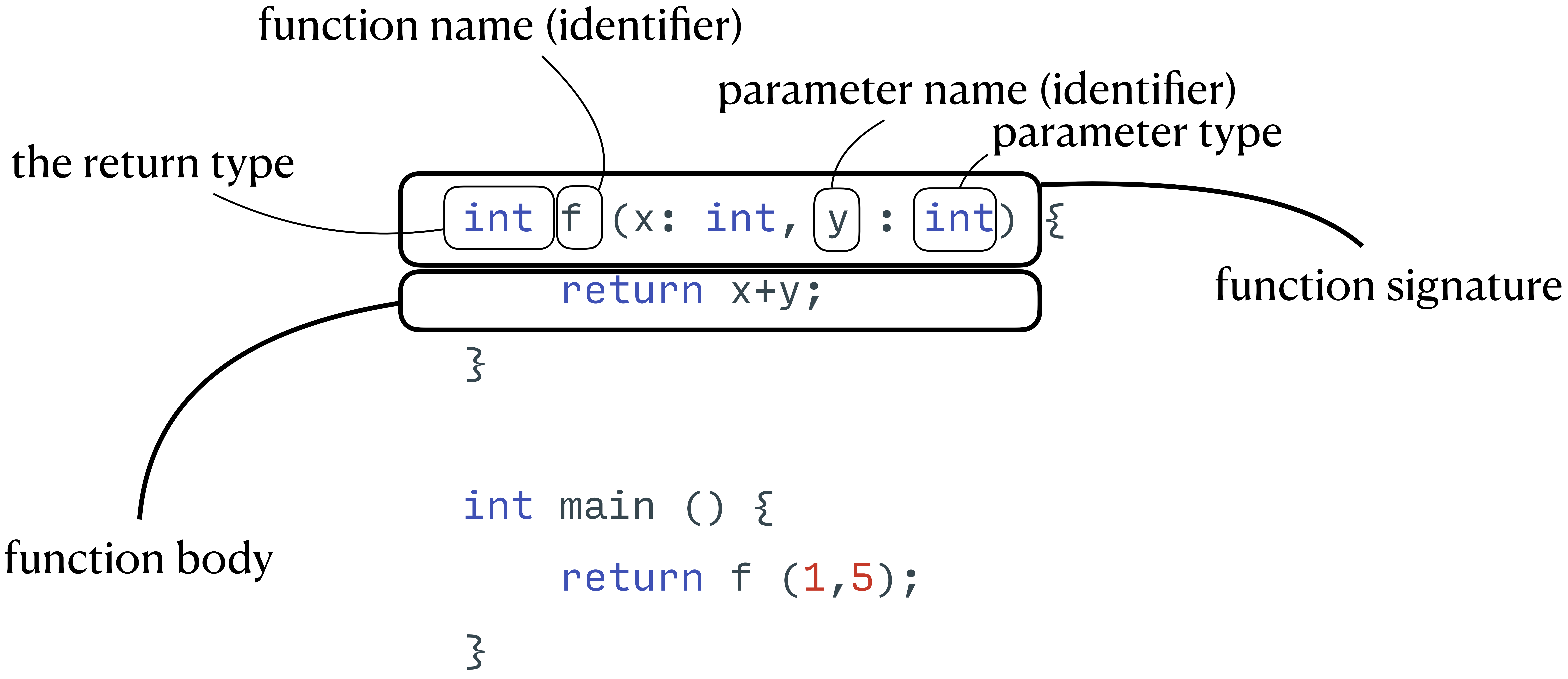
Function example

```
int f (x: int, y : int) {
    return x+y;
}

int main () {
    return f (1,5);
}
```

Function example



```
int f (x: int, y : int) {
    return x+y;
}


int main () {
    return f (1,5);
}
```



```
int f (x: int, y : int) {
    return x+y;
}


int main () {
    return f (1,5);
}
```

Comma expression example

```
int main () {
    var _o = get_stdout ();
    for (var i: int = 0, j: int = 9; i < 10 ; i = i+1, j=j-1) {
        output_string (int_to_string (i), _o);
        output_string (" ", _o);
        output_string (int_to_string (j), _o);
        output_string ("\n", _o);
   }
    return 0;
}
```

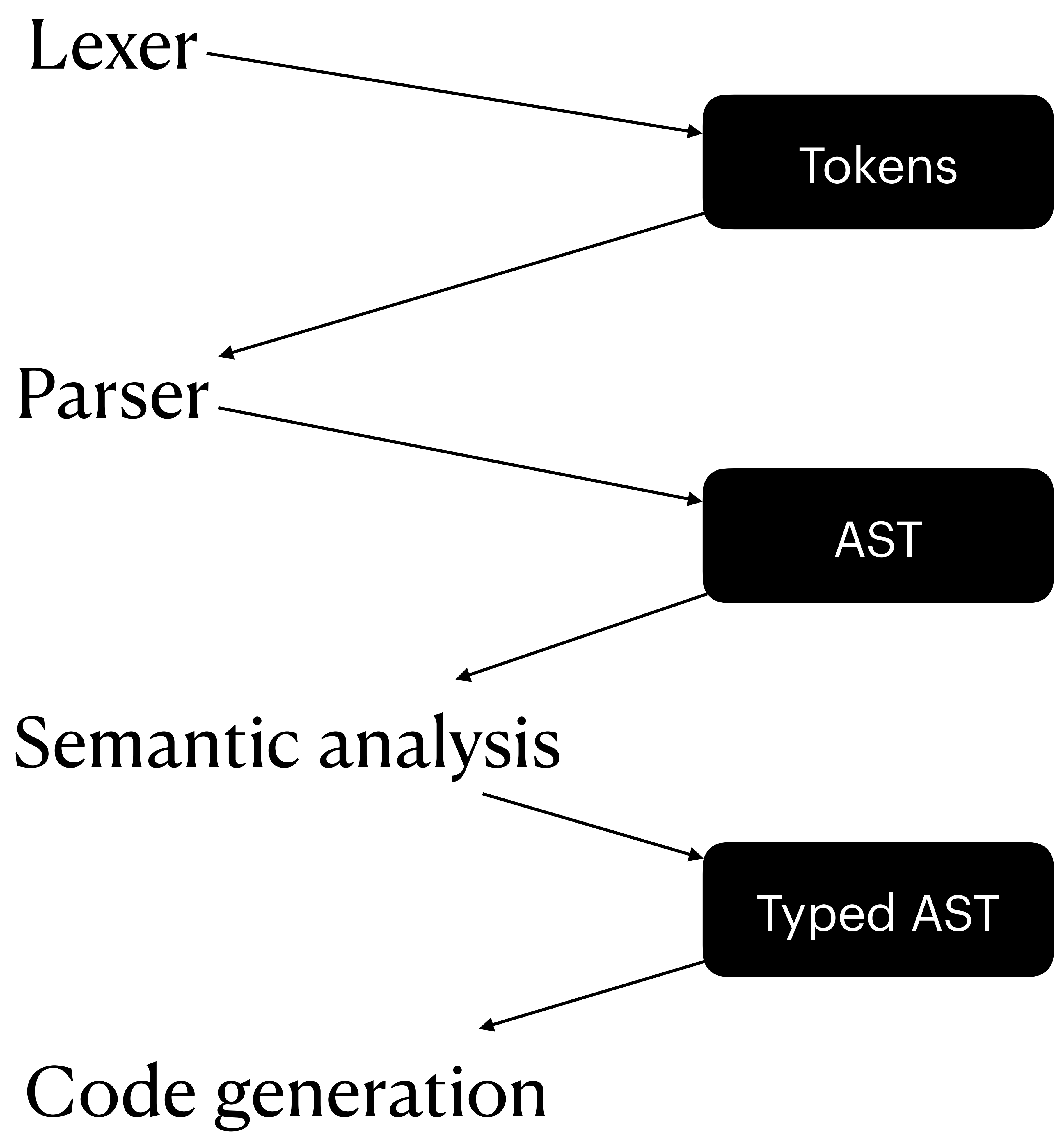
Comma expression example

```
int main () {
    var _o = get_stdout ();
    for (var i: int = 0, j: int = 9; i < 10 ; i = i+1, j=j-1) {
        output_string (int_to_string (i), _o);
        output_string (" ", _o);
        output_string (int_to_string (j), _o);
        output_string ("\n", _o);
    }
    return 0;
}
```

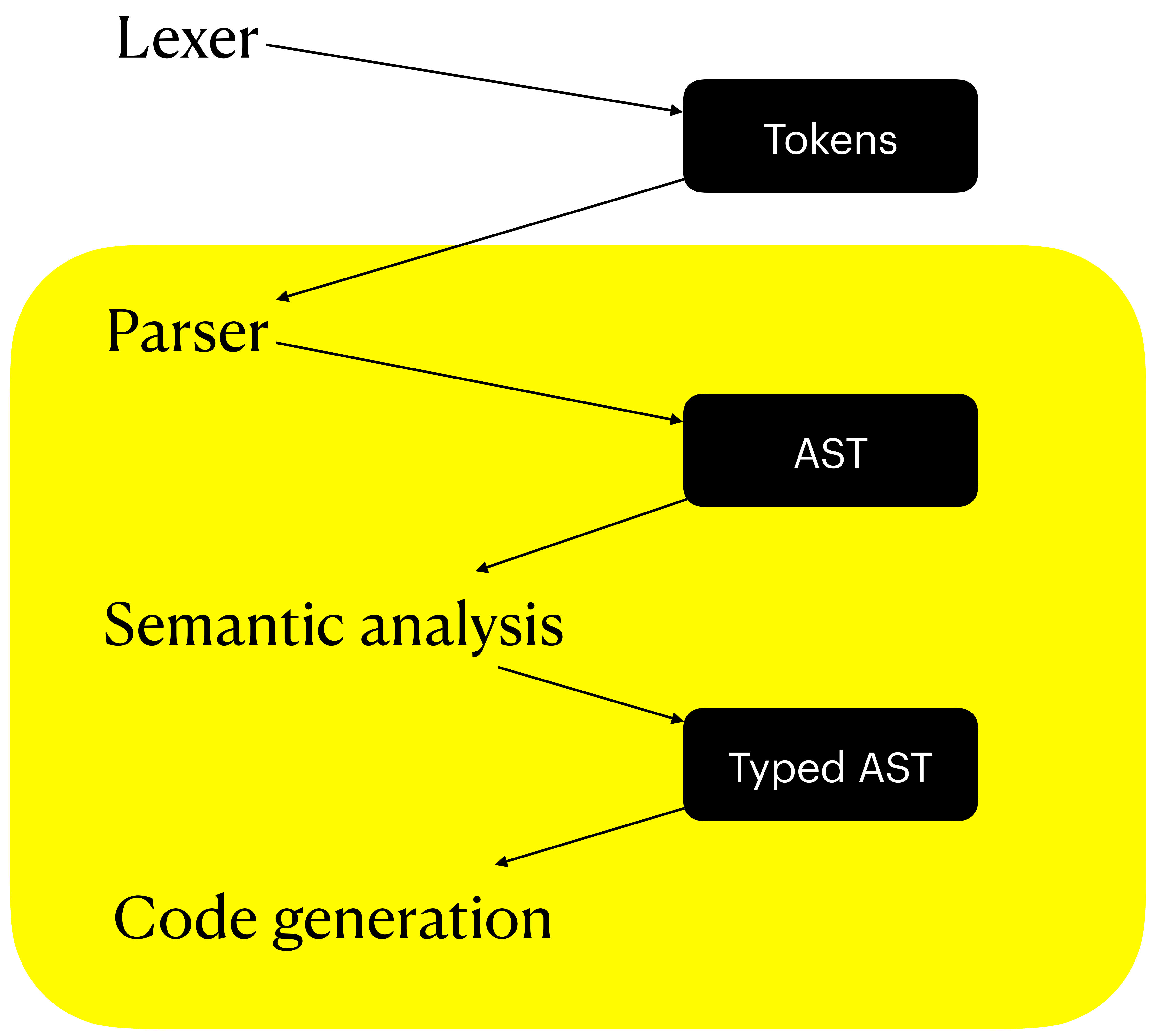
comma sequences two expressions

loop update expression

Compiler pipeline



What needs to change?



No changes in Lexer/Tokens

Parts that require modification

AST changes

- Program is a list of functions
- A function includes all the information about the function
  - function name; return type; argument names and types; body; location
- Comma expression needs to be added to the AST
  - similar to binop: includes left and right hand sides
- Note:
  - No AST codebase is provided in this assignment!
    - Design and implement it yourselves
    - Don’t forget the pretty printing