Semantic Analysis for MicroC

Letterio Galletta

Remember MicroC

MicroC is a sublanguage of C

Many simplifications have been made compared to real C

- datatypes: only int and char variables, arrays, and pointers
- no structs, unions, doubles, function pointers, ...
- no initializers in variable declarations
- Functions can return only int, char, void, bool
- No pointer arithmetic
- Pointers and arrays are not interchangeable
- No dynamic allocation of memory
- No automatic type coercion

Scoping rules

MicroC is a language with static scoping:

- Blocks can be nested
- A variable x in a inner block hides a variables with the same name in an outer block
- There is a global scope where global variables are
- Function declarations cannot be nested
- No function overloading

Main function

MicroC does not support separate compilation:

- Each source file must provide the definition of the main function
- The signature of the main function can be

```
void main()   // it returns no value

int main()    // it returns an integer
```

Library functions

We assume to have two library functions:

- void print(int n) // print an integer to standard output
- int getint() // read an integer from the standard input

Types (1)

The types are given by the following grammar

$$T ::= \text{int} \mid \text{bool} \mid \text{char} \mid \text{void} \mid T^* \mid T[]$$

There are no type coercion:

- Booleans and characters cannot be converted to integers
- Arrays and pointers are different types

Types (2)

- Arithmetic operators expect only integers
- Logical operators expect only boolean values
- Dereference operators expects a pointer
- _[_] operator expects an array and an integer as index
- Guards in if and while expect boolean values

Other semantic rules

- Array should have a size of at least 1
- Array cannot be at the left of an assignment, i.e., array1 = arrayB, is not allowed
- void variables are not allowed
- No partial application of functions
- Functions are not expressible values and there are no function pointers
- Functions cannot return pointers or arrays
- No multi-dimensional arrays