Programming in the Past

CMPT 331 - Spring 2023 | Dr. Labouseur

Josh Seligman | joshua.seligman1@marist.edu

February 1, 2023

1 Crafting a Compiler

1.1 Exercise 1.11

The Measure Of Software Similarity (MOSS) [SWA03] tool can detect similarity of programs written in a variety of modern programming languages. Its main application has been in detecting similarity of programs submitted in computer science classes, where such similarity may indicate plagiarism (students, beware!). In theory, detecting equivalence of two programs is undecidable, butMOSS does a very good job of finding similarity in spite of that limitation.
Investigate the techniques MOSS uses to find similarity. How does MOSS differ from other approaches for detecting possible plagiarism?

The MOSS similarity detection begins by stripping whitespace and identifiers from the code to put it in a position that is language independent for the rest of the algorithm to use. Next, it uses hashing to create what are known as document fingerprints, which represent the basic structure of the code. Lastly, MOSS compares these fingerprints with the fingerprints of other documents, which marks the similarity. This is different from other approaches as it makes sure that whitespace and single word matches do not impact the algorithm results as well as the position of the code (order in which functions are defined). Thus, the MOSS detection algorithm compares the true meaning of the code rather than the superficial characteristics like variable names and whitespace.

Sources: http://theory.stanford.edu/ aiken/publications/papers/sigmod03.pdf,
https://yangdanny97.github.io/blog/2019/05/03/MOSS

1.2 Exercise 3.1

What token sequence is produced? For which tokens must extra information be returned in addition to the token code?

```
main(){
    const float payment = 384.00;
    float bal;
    int month = 0;
    bal=15000;
    while (bal>0){
        printf("Month: %2d Balance: %10.2f\n", month, bal);
        bal=bal-payment+0.015*bal;
        month=month+1;
    }
}
```

Token	Additional Information
Identifier	main
LParen	N/A
RParen	N/A
LBrace	N/A
Const	N/A
Float	N/A
Identifier	payment
AssignmentOperator	N/A
Number	384.00
Semicolon	N/A
Float	N/A
Identifier	bal
Semicolon	N/A
Int	N/A
Identifier	month
AssignmentOperator	N/A
Number	0
Semicolon	N/A
Identifier	bal
AssignmentOperator	N/A
Number	1500
Semicolon	N/A
While	N/A
LParen	N/A
Identifier	bal
GreaterThan	N/A
Number	0
LBrace	N/A
Identifier	printf
LParen	N/A
Quote	N/A
Characters	Month: %2d Balance: %10.2f\n
Quote	N/A
Comma	N/A
Identifier	month
Comma	N/A
Identifier	bal
RParen	N/A
Semicolon	N/A
Identifier	bal
AssignmentOperator	N/A
Identifier	bal
Subtraction	N/A
Identifier	payment
Addition	N/A
Number	0.015
Multiply	N/A
Identifier	bal

Semicolon	N/A
Identifier	month
AssignmentOperator	N/A
Identifier	month
Addition	N/A
Number	1
Semicolon	N/A
RBrace	N/A
RBrace	N/A

2 Dragon

2.1 Exercise 1.1.4

A compiler that translates a high-level language into another high-level language is called a source-to-source translator. What advantages are there to using C as a target language for a compiler?

Using C as a target language for a compiler can be effective as you just have to translate the language to C rather than dealing with any platform-specifics. Also, C compilers are extremely common, so the new language can be run on any computer that can run C, which increases portability of the new language.

2.2 Exercise 1.6.1

For the block-structured C code of Fig. 1.13(a), indicate the values assigned to w, x, y, and z.

```
int w, x, y, z;
int i = 4; int j = 5;
{
    int j = 7;
    i = 6;
    w = i + j;
}
x = i + j;
{
    int i = 8;
    y = i + j;
}
z = i + j;
```

w gets assigned 13 because the j within the first block scope has a value of 7 and the i defined in the main scope gets assigned the value of 6.

x is assigned 11 because the main scope i is assigned 6 and the j is unaffected because the j in the first scope is a separate variable.

y is assigned 13 because the new i variable is 8 and j is still 5.

z is assigned 11 because i and j have not been changed since the assignment to x