Brendon Boldt
January 28, 2016
Lab 1

Crafting a Compiler

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, but MOSS 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?

MOSS works by detecting similarities in source code not by comparing source code to source code, but by comparing token streams to token streams. When looking at just the source code, changing variable names and switching declaration order can make things seem superficially different. Yet looking for matching tokens and lines produced by lexing gives a more objective view of the program structure that is not affected by changed variables names and such.

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

Tokens needing additional information underlined.

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 ; } }

Dragon

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?

C is a good target language because it is one of the most common and widely implemented programming languages in existence. This means that almost every platform/architecture will have a well-built compiler that can compile C into executable code for that platform. As a result, any code written in the original source language can be compiled into C by the source-to-source compiler and then compiled into the executable code by the platform-specific C compiler.

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 == 13
x == 11
y == 13
z == 13
```