

TDT4205 - Øving 4

Daniel Romanich

March 2018

1 Theory

1.1 Three address code

The three address code is a language that is very similar to assembly. It is called three address code because it relies on three registers to process data with. Three address code fits very well as an implementation into a stack machine due to its registers. Since every operation in TAC at most has 2 operands it will never **HAVE** to use more than two registers at a time. This allows for the last register to be used as a stack pointer. This allows TAC to always keep count of the top-of-stack position, and thus every operation can be executed by pushing/popping elements to the stack and updating the register to point at the top. The reason to why its discussed in compiler construction is that it is very easy to implement a stack machine using TAC, and therefore explains the principles in a good way.

1.2 C to TAC

Program:

```
1 int main(void) {  
2     int a;  
3     a = 2;  
4     int b;  
5     b = 2 * a + 3;  
6     int c;  
7     if (b == 7) {  
8         c = 4;  
9     } else {  
10        c = 6;  
11    }  
12 }
```

```
1     t0 = 2; //the variable a  
2     t1 = 2; //the constant to be multiplied by a  
3     t2 = t0 * t1; //2 * a  
4     t3 = 3; //the constant to be added to 2 * a  
5     t4 = t3 + t2; //b  
6     t5 = 7; //the constant we compare b to  
7     t6 = (t5 == t4); //store the comparison of b and 7  
8     ifFalse t6 goto L2; //if b == 7 go to the
```

```
9 L1:
10     t7 = 4; //assign the value 4 to c
11     goto L3; //jump over the 'else'
12 L2:
13     t7 = 6; //assign the value 6 to c
14 L3:
15     //end of code
```

2 Programming

All of the code is provided together with this report. The only difference in output when comparing it to the correct outputs provided is the sequencing. In my code, parameters are also sequenced, meaning the local variables get a higher sequencing number than in the provided solution. Since the powerpoint slides said to number the parameters I kept the difference, but if this is wrong it has a one line fix (prevent the sequence from counting when adding parameters).