

## Compilerbau - Wintersemester 2021/22

### Theoretisches Übungsblatt 9

Besprechung der Aufgaben am 07.01.22 ab 14:30 Uhr in 25.12.02.55 und gleichzeitig online per BBB  
Fragen an [lukas.lang@hhu.de](mailto:lukas.lang@hhu.de)  
Die Bearbeitung ist freiwillig.

#### Aufgabe 9.1

Verwenden Sie die Attributgrammatik aus der Vorlesung (IR Folie 16) um den arithmetischen Ausdruck  $a + -(b + c)$  sowie den Ausdruck  $((x+y)-((x+y) * (x-y))) + ((x+y) * (x-y))$  zu parsen und übersetzen Sie beide jeweils in :

- (a) Einen Syntaxbaum
- (b) 0-Adress-Code (Bytecode)
- (c) Einen DAG
- (d) 3-Adress-Code
- (e) Tripel (bzw. Indirekte Tripel)
- (f) Quadrupel
- (g) SSA

Für welche Situationen eignet sich welche Darstellung? Geben Sie Beispiele an.

**Hinweis:** In der Attributgrammatik aus der Vorlesung befindet sich keine Regel für `**`. Ein `**` können Sie analog zu `+` behandeln.

#### Aufgabe 9.2

Erzeugen Sie Drei-Adress-Code für folgenden Codeschnipsel:

```
x=a[b[i]][3] + b[j]
```

Die Variablen a und b sind 32Bit Integer Arrays der Länge `int[10][4]` und `int[3]` Verwenden Sie die Attributgrammatik aus der Vorlesung (IR Folie 17)

#### Aufgabe 9.3

Erzeugen Sie Drei-Adress-Code für folgenden Codeschnipsel:

```
while (x < y && x > 200) {  
    if (a+1 == b || a < 0) {  
        x = 42;  
    }  
}
```

Die Variablen a, b, x und y sind vom Typ Integer. Verwenden Sie die Attributgrammatik aus der Vorlesung (IR Folie 19) .