

Compiler Construction: Assignment 3

Fabian Krause

December 6, 2021

Assignment 3: Register Allocation for \mathcal{L}_{Var}

1. Liveness Analysis
2. Inteference Graph
3. Graph Coloring
4. Allocate Registers/Assign Homes
5. Patch Instructions & Prelude and Conclusion

1. Liveness Analysis

A variable is live at a program point k if its current value is used at some later point in the program:

$$L_{before}(k) = (L_{after}(k) - W(k)) \cup R(k)$$

Auxiliary functions:

- ▶ Arg Locations
- ▶ Read/Write Locations

2. Interference Graph

Two variables interfere if they are live at the same time.

1. If instruction I_k is a move instruction of the form `movq s, d`, then for every $v \in L_{\text{after}}(k)$, if $v \neq d$ and $v \neq s$, add the edge (d, v) .
2. For any other instruction I_k , for every $d \in W(k)$ and every $v \in L_{\text{after}}(k)$, if $v \neq d$, add the edge (d, v) .

3. Graph Coloring

A graph G has a k -coloring, if each node in G can be assigned a number in $\{0, \dots, k - 1\}$ such that any two adjacent nodes are assigned two distinct numbers.

3.1. Precoloring

- ▶ How to handle interferences between variables and registers?
- ▶ Fixed mapping $\{\text{\%rbp}, \text{\%rsp}, \dots\} \rightarrow \{-3, -2, \dots, 12\}$ (first caller-saved, then callee-saved):

$$\{\text{\%rbp} \mapsto -3, \text{\%rsp} \mapsto -2, \text{\%rax} \mapsto -1, \text{\%rcx} \mapsto 0, \dots\}$$

- ▶ Assign registers in interference graph their color

3.2. Coloring Algorithm

```
1  $W \leftarrow \text{vertices}(G)$ 
2 while  $W \neq \emptyset$  do
3   pick a vertex  $u$  from  $W$  with the highest saturation,
4     breaking ties randomly
5   find the lowest color  $c$  that is not in
      $\{\text{color}[v] : v \in \text{adjacent}(u)\}$ 
6    $\text{color}[u] \leftarrow c$ 
7    $W \leftarrow W - \{u\}$ 
```

Use priority queue for picking vertex with highest saturation

4. Allocate Registers/Assign Homes

Integration into the compiler:

- ▶ Build interference graph
- ▶ Create coloring
- ▶ Assign colors to registers/stack:
 - ▶ Use inverse of mapping
 $\{\%rbp, \%rsp, \dots\} \rightarrow \{-3, -2, \dots, 12\}$ for $-3 \leq c \leq 12$
 - ▶ Create new stack locations for colors $c > 12$

5. Patch Instructions & Prelude and Conclusion

- ▶ Delete trivial `movq` instructions
- ▶ Store & restore callee-saved registers

Questions?