$\boxed{\text{Lec 10}}$ Calling Conventions

IR for calling convention

Goal: args are pure, args evaled left to right

higher level $\qquad d \leftarrow f(s_1, \dots, s_n)$

lower level $\qquad$ CALL f

x86-64 calling convention

first 6 args* go on certain regs, others go stack
└ except floating point args go on %xmm0 - %xmm7

stack when calling func

```
            _____________
                  ⋮
            _____________
16(%rsp)        arg9
            _____________
 8(%rsp)        arg8
            _____________
   %rsp       ret addr        ↑ Caller frame
            _____________
```

%rdi	arg1	%rdx	arg3	%r8	arg5
%rsi	arg2	%rcx	arg4	%r9	arg6

Good practices
- 8 byte per arg on stack
- Statically precompute frame size
- % rbp aligned 0 mod 16 before call, 8 mod 16 on entry

Renaming

```
res₀      rax     return     ┬
arg₁      rdi     ┬          │
arg₂      rsi     │          │
arg₃      rdx     args       caller-saved
arg₄      rcx     │          │
arg₅      r8      ┴          │
arg₆      r9                 │
ler₇      r10                │
ler₈      r11                ┴
lee₉      rbx     ┬
lee₁₀     rbp     │
lee₁₁     r12     callee-saved
lee₁₂     r13     │
lee₁₃     r14     │
lee₁₄     r15     ┴
          rsp     stack ptr
```

Reg alloc

Caller-saved & arg regs can get overwritten

$$\frac{l:\ \text{call } f \qquad \text{caller-save}(r)}{\text{def}(l, r)} \qquad\qquad \frac{l:\ \text{ret } s \qquad \text{callee-save}(r)}{\text{use}(l, r)}$$

Suppose $\qquad d \leftarrow f(s_1, s_2, s_3)$

$\rightsquigarrow$

$arg_3 \leftarrow s_3$
$arg_2 \leftarrow s_2$
$arg_1 \leftarrow s_1$
call f
$d \leftarrow res_0$

← if t used here then it interferes with all the caller-saved regs

At start of func, try not hold precolored regs for too long

f:
$\quad x \leftarrow arg_1$
$\quad y \leftarrow arg_2$
$\quad z \leftarrow arg_3$

Saving stuff

Standard: save all lee* onto stack & restore them at end
└ but then they live throughout the func

Spill techniques:
1. reg alloc on caller-saved first, spill to caller-saved if necessary, then spill to stack
2. move them to temp and move them back, let reg alloc decide
3. copy propagation...?

e.g. (2) $\qquad t_1 \leftarrow lee_9$
$\qquad\qquad t_2 \leftarrow lee_{10}$
$\qquad\qquad\quad \vdots$
$\qquad\qquad lee_{10} \leftarrow t_2$
$\qquad\qquad lee_9 \leftarrow t_1$