CS 4120
Introduction to Compilers

Ross Tate
Cornell University

Lecture 33: Loop Optimizations II

Induction-Variable Strength Reduction

```
while (i < 10) {
    i = i + 2;
    m = i * 8;
    use(m + 6);
}
```

```
m = i * 8;
while (i < 10) {
    i = i + 2;
    m = m + 16;
    use(m + 6);
}
```


Loop Unrolling

```
m = i * 8;
while (i < 10) {
    i = i + 2;
    m = m + 16;
    use(m + 6);
}
```

```
m = i * 8;
while (i < 10-2*1) {
    i = i + 2;
    m = m + 16;
    use(m + 6);
    i = i + 2;
    m = m + 16;
    use(m + 6);
}
```


Common-Subexpression Elimination

```
a := i;
b := a * 2;
if (c)
    d := a + b;
    b := 0;
else
    d := i + b;
e := b * 3;
f := b * 3;
i := i * 2;
return e + f + i*2;
```



```
Foo(n)
    if (n > 0)
        return f(n-1) + f(n-1)
    else
        return 0
```