15-411: Induction Variables

Jan Hoffmann

Example

Check if an array is sorted

```
bool is_sorted(int[] A, int n)
//@requires 0 <= n && n <= \length(A);
{
  for (int i = 0; i < n-1; i++)
      //@loop_invariant 0 <= i;
    if (A[i] > A[i+1]) return false;
  return true;
}
```

```
\mathsf{is\_sorted}(A, n) :
       i_0 \leftarrow 0
       goto loop(i_0)
loop(i_1):
       t_0 \leftarrow n-1
       if (i_1 \ge t_0) goto rtrue
       t_1 \leftarrow 4 * i_1
       t_2 \leftarrow A + t_1
       t_3 \leftarrow M[t_2]
       t_4 \leftarrow i_1 + 1
       t_5 \leftarrow 4 * t_4
       t_6 \leftarrow A + t_5
       t_7 \leftarrow M[t_6]
       if (t_3 > t_7) goto rfalse
       i_2 \leftarrow i_1 + 1
       goto loop(i_2)
rtrue:
       return 1
rfalse:
       return 0
```

Translation to SSA Form (without array bound checks)

Basic induction variable

```
\mathsf{is\_sorted}(A, n) :
       i_0 \leftarrow 0
       goto loop(i_0)
loop(i_1):
       t_0 \leftarrow n-1
       if (i_1 \ge t_0) goto rtrue
       t_1 \leftarrow 4 * i_1
       t_2 \leftarrow A + t_1
       t_3 \leftarrow M[t_2]
       t_4 \leftarrow i_1 + 1
       t_5 \leftarrow 4 * t_4
       t_6 \leftarrow A + t_5
       t_7 \leftarrow M[t_6]
       if (t_3 > t_7) goto rfalse
       i_2 \leftarrow i_1 + 1
       goto loop(i_2)
rtrue:
       return 1
rfalse:
       return 0
```

Translation to SSA Form (without array bound checks)

Basic induction variable

Derived induction variable

```
\mathsf{is\_sorted}(A, n) :
       i_0 \leftarrow 0
       goto loop(i_0)
loop(i_1):
       t_0 \leftarrow n-1
       if (i_1 \ge t_0) goto rtrue
      t_1 \leftarrow 4 * i_1
       t_2 \leftarrow A + t_1
       t_3 \leftarrow M[t_2]
      t_4 \leftarrow i_1 + 1
       t_5 \leftarrow 4 * t_4
       t_6 \leftarrow A + t_5
       t_7 \leftarrow M[t_6]
       if (t_3 > t_7) goto rfalse
      i_2 \leftarrow i_1 + 1
       goto loop(i_2)
rtrue:
       return 1
rfalse:
       return 0
```

Translation to SSA Form (without array bound checks)

$$\begin{array}{c} \mathrm{is_sorted}(A,n): \\ i_0 \leftarrow 0 \\ \mathrm{goto} \ \mathrm{loop}(i_0) \\ \mathrm{loop}(i_1): \\ t_0 \leftarrow n-1 \\ \mathrm{if} \ (i_1 \geq t_0) \ \mathrm{goto} \ \mathrm{rtrue} \\ t_1 \leftarrow 4*i_1 \\ t_2 \leftarrow A+t_1 \\ t_3 \leftarrow M[t_2] \\ t_4 \leftarrow i_1+1 \\ t_5 \leftarrow 4*t_4 \\ t_6 \leftarrow A+t_5 \\ t_7 \leftarrow M[t_6] \\ \mathrm{if} \ (t_3 > t_7) \ \mathrm{goto} \ \mathrm{rfalse} \\ i_2 \leftarrow i_1+1 \\ \mathrm{goto} \ \mathrm{loop}(i_2) \\ \end{array}$$

$$i_0 \leftarrow 0$$
 $goto loop(i_0)$
 $loop(i_1)$:

 $t_0 \leftarrow n-1$

if $(i_1 \ge t_0)$ goto rtrue

 $t_1 \leftarrow 4 * i_1$

 $t_2 \leftarrow A + t_1$

 $t_3 \leftarrow M[t_2]$

 $t_4 \leftarrow i_1 + 1$

 $t_5 \leftarrow 4 * t_4$

 $t_6 \leftarrow A + t_5$

 $t_7 \leftarrow M[t_6]$

if $(t_3 > t_7)$ goto rfalse

 $i_2 \leftarrow i_1 + 1$

goto $loop(i_2)$

Common subexpression elimination applies

$$i_0 \leftarrow 0$$
 goto $loop(i_0)$

 $loop(i_1)$:

$$t_0 \leftarrow n-1$$

if $(i_1 \ge t_0)$ goto rtrue

$$t_1 \leftarrow 4 * i_1$$

$$t_2 \leftarrow A + t_1$$

$$t_3 \leftarrow M[t_2]$$

$$t_4 \leftarrow i_1 + 1$$

$$t_5 \leftarrow 4 * t_4$$

$$t_6 \leftarrow A + t_5$$

$$t_7 \leftarrow M[t_6]$$

if $(t_3 > t_7)$ goto rfalse

$$i_2 \leftarrow i_1 + 1$$

goto $\mathsf{loop}(i_2)$

Common subexpression elimination applies

$$i_0 \leftarrow 0$$
 goto $loop(i_0)$

 $loop(i_1)$:

$$t_0 \leftarrow n-1$$

if $(i_1 \ge t_0)$ goto rtrue

$$t_1 \leftarrow 4 * i_1$$

$$t_2 \leftarrow A + t_1$$

$$t_3 \leftarrow M[t_2]$$

$$t_4 \leftarrow i_1 + 1$$

$$t_5 \leftarrow 4 * t_4$$

$$t_6 \leftarrow A + t_5$$

$$t_7 \leftarrow M[t_6]$$

if $(t_3 > t_7)$ goto rfalse

$$i_2 \leftarrow i_1 + 1$$

goto $loop(i_2)$

Preserve induction variable i

Common subexpression elimination applies

```
is\_sorted(A, n):
       i_0 \leftarrow 0
       goto loop(i_0)
loop(i_1):
       t_0 \leftarrow n-1
       if (i_1 \geq t_0) goto rtrue
       t_1 \leftarrow 4 * i_1
       t_2 \leftarrow A + t_1
       t_3 \leftarrow M[t_2]
       t_4 \leftarrow i_1 + 1
       t_5 \leftarrow 4 * t_4
       t_6 \leftarrow A + t_5
       t_7 \leftarrow M[t_6]
       if (t_3 > t_7) goto rfalse
       i_2 \leftarrow t_4
       goto loop(i_2)
```

```
is\_sorted(A, n):
       i_0 \leftarrow 0
       goto loop(i_0)
loop(i_1):
       t_0 \leftarrow n-1
       if (i_1 \geq t_0) goto rtrue
       t_1 \leftarrow 4 * i_1
       t_2 \leftarrow A + t_1
       t_3 \leftarrow M[t_2]
       t_4 \leftarrow i_1 + 1
       t_5 \leftarrow 4 * t_4
       t_6 \leftarrow A + t_5
       t_7 \leftarrow M[t_6]
       if (t_3 > t_7) goto rfalse
       i_2 \leftarrow t_4
       goto loop(i_2)
```

$$i_0 \leftarrow 0$$

$$\operatorname{goto} \operatorname{loop}(i_0)$$

$$\operatorname{loop}(i_1):$$

$$t_0 \leftarrow n-1$$

$$\operatorname{if}\ (i_1 \geq t_0) \operatorname{goto} \operatorname{rtrue}$$

$$t_1 \leftarrow 4*i_1$$

$$t_2 \leftarrow A+t_1$$

$$t_3 \leftarrow M[t_2]$$

$$i_2 \leftarrow i_1+1$$

$$t_5 \leftarrow 4*i_2$$

$$t_6 \leftarrow A+t_5$$

$$t_7 \leftarrow M[t_6]$$

$$\operatorname{if}\ (t_3 > t_7) \operatorname{goto} \operatorname{rfalse}$$

$$\operatorname{goto} \operatorname{loop}(i_2)$$

Consider induction variable t1

$$t_1 \leftarrow 4 * i_1$$

Idea: compute t1 from a previous iteration of t1

Introduce new variable j = 4*i

$$\begin{array}{c} \mathrm{is_sorted}(A,n): \\ i_0 \leftarrow 0 \\ \mathrm{goto} \ \mathrm{loop}(i_0) \\ \mathrm{loop}(i_1): \\ t_0 \leftarrow n-1 \\ \mathrm{if} \ (i_1 \geq t_0) \ \mathrm{goto} \ \mathrm{rtrue} \\ \hline t_1 \leftarrow 4*i_1 \\ t_2 \leftarrow A+t_1 \\ t_3 \leftarrow M[t_2] \\ i_2 \leftarrow i_1+1 \\ t_5 \leftarrow 4*i_2 \\ t_6 \leftarrow A+t_5 \\ t_7 \leftarrow M[t_6] \\ \mathrm{if} \ (t_3 > t_7) \ \mathrm{goto} \ \mathrm{rfalse} \\ \\ \mathrm{goto} \ \mathrm{loop}(i_2) \\ \end{array}$$

Consider induction variable t1

$$t_1 \leftarrow 4 * i_1$$

Idea: compute t1 from a previous iteration of t1

Introduce new variable j = 4*i

$$egin{aligned} \mathsf{is_sorted}(A,n): \\ i_0 \leftarrow 0 \\ j_0 \leftarrow 4*i_0 \\ \mathsf{goto} \ \mathsf{loop}(i_0,j_0) \end{aligned}$$

$\mathsf{loop}(i_1,j_1):$

$$t_0 \leftarrow n-1$$
 if $(i_1 \ge t_0)$ goto rtrue

$$t_1 \leftarrow j_1$$

$$t_2 \leftarrow A + t_1$$

$$t_3 \leftarrow M[t_2]$$

$$i_2 \leftarrow i_1 + 1$$

$$j_2 \leftarrow 4 * i_2$$

$$t_5 \leftarrow 4 * i_2$$

$$t_6 \leftarrow A + t_5$$

$$t_7 \leftarrow M[t_6]$$

if
$$(t_3 > t_7)$$
 goto rfalse goto loop (i_2, j_2)

$$@ensures j_0 = 4 * i_0$$

$$@ requires j_1 = 4 * i_1$$

@assert
$$j_1 = 4 * i_1$$

$$@ensures j_2 = 4 * i_2$$

```
\mathsf{is\_sorted}(A, n) :
       i_0 \leftarrow 0
       j_0 \leftarrow 4 * i_0
       goto loop(i_0, j_0)
loop(i_1, j_1):
       t_0 \leftarrow n-1
       if (i_1 \ge t_0) goto rtrue
       t_1 \leftarrow j_1
       t_2 \leftarrow A + t_1
       t_3 \leftarrow M[t_2]
       i_2 \leftarrow i_1 + 1
       j_2 \leftarrow 4 * i_2
       t_5 \leftarrow 4 * i_2
       t_6 \leftarrow A + t_5
       t_7 \leftarrow M[t_6]
       if (t_3 > t_7) goto rfalse
       goto loop(i_2, j_2)
```

$$\begin{array}{l} \operatorname{is_sorted}(A,n): \\ i_0 \leftarrow 0 \\ j_0 \leftarrow 0 \\ \operatorname{goto} \operatorname{loop}(i_0,j_0) \\ \operatorname{loop}(i_1,j_1): \\ t_0 \leftarrow n-1 \\ \operatorname{if}\ (i_1 \geq t_0) \operatorname{goto} \operatorname{rtrue} \\ t_1 \leftarrow j_1 \\ t_2 \leftarrow A + t_1 \\ t_3 \leftarrow M[t_2] \\ i_2 \leftarrow i_1 + 1 \\ j_2 \leftarrow j_1 + 4 \\ t_5 \leftarrow 4 * i_2 \\ t_6 \leftarrow A + t_5 \\ t_7 \leftarrow M[t_6] \\ \operatorname{if}\ (t_3 > t_7) \operatorname{goto} \operatorname{rfalse} \\ \operatorname{goto} \operatorname{loop}(i_2,j_2) \\ \end{array}$$

 $\mathsf{is_sorted}(A, n) :$ $i_0 \leftarrow 0$ $j_0 \leftarrow 4 * i_0$ goto $loop(i_0, j_0)$ $\mathsf{loop}(i_1, j_1)$: $t_0 \leftarrow n-1$ if $(i_1 \ge t_0)$ goto rtrue $t_1 \leftarrow j_1$ $t_2 \leftarrow A + t_1$ $t_3 \leftarrow M[t_2]$ $i_2 \leftarrow i_1 + 1$ $j_2 \leftarrow 4 * i_2$ $t_5 \leftarrow 4 * i_2$ $t_6 \leftarrow A + t_5$ $t_7 \leftarrow M[t_6]$ if $(t_3 > t_7)$ goto rfalse goto $loop(i_2, j_2)$

 $is_sorted(A, n)$: $i_0 \leftarrow 0$ $j_0 \leftarrow 0$ goto $loop(i_0, j_0)$ $\mathsf{loop}(i_1, j_1)$: $t_0 \leftarrow n-1$ if $(i_1 \ge t_0)$ goto rtrue $t_1 \leftarrow j_1$ $t_2 \leftarrow A + t_1$ $t_3 \leftarrow M[t_2]$ $i_2 \leftarrow i_1 + 1$ $j_2 \leftarrow j_1 + 4$ $t_5 \leftarrow 4 * i_2$ $t_6 \leftarrow A + t_5$ $t_7 \leftarrow M[t_6]$ if $(t_3 > t_7)$ goto rfalse goto $loop(i_2, j_2)$

Qensures $j_0 = 4 * i_0$ @requires $j_1 = 4 * i_1$ @assert $j_1 = 4 * i_1$ @ensures $j_2 = 4 * i_2$

 $\mathsf{is_sorted}(A, n)$: $i_0 \leftarrow 0$ $j_0 \leftarrow 4 * i_0$ goto $loop(i_0, j_0)$ $\mathsf{loop}(i_1, j_1)$: $t_0 \leftarrow n-1$ if $(i_1 \geq t_0)$ goto rtrue $t_1 \leftarrow j_1$ $t_2 \leftarrow A + t_1$ $t_3 \leftarrow M[t_2]$ $i_2 \leftarrow i_1 + 1$ $j_2 \leftarrow 4 * i_2$ $t_5 \leftarrow 4 * i_2$ $t_6 \leftarrow A + t_5$ $t_7 \leftarrow M[t_6]$ if $(t_3 > t_7)$ goto rfalse goto $loop(i_2, j_2)$

 $j_0 = 4 * i_0 = 0$

 $is_sorted(A, n)$: $i_0 \leftarrow 0$ $j_0 \leftarrow 0$ $@ensures j_0 = 4 * i_0$ goto $loop(i_0, j_0)$ $\mathsf{loop}(i_1, j_1)$: @requires $j_1 = 4 * i_1$ $t_0 \leftarrow n-1$ if $(i_1 \ge t_0)$ goto rtrue @assert $j_1 = 4 * i_1$ $t_1 \leftarrow j_1$ $t_2 \leftarrow A + t_1$ $t_3 \leftarrow M[t_2]$ $i_2 \leftarrow i_1 + 1$ $j_2 \leftarrow j_1 + 4$ @ensures $j_2 = 4 * i_2$ $t_5 \leftarrow 4 * i_2$ $t_6 \leftarrow A + t_5$ $t_7 \leftarrow M[t_6]$ if $(t_3 > t_7)$ goto rfalse goto $loop(i_2, j_2)$

$$j_2 = 4 * i_2 = 4 * (i_1 + 1) = 4 * i_1 + 4 = j_1 + 4$$

```
\mathsf{is\_sorted}(A, n) :
       i_0 \leftarrow 0
       j_0 \leftarrow 0
       goto loop(i_0, j_0)
\mathsf{loop}(i_1, j_1):
       t_0 \leftarrow n-1
       if (i_1 \ge t_0) goto rtrue
       t_1 \leftarrow j_1
       t_2 \leftarrow A + t_1
       t_3 \leftarrow M[t_2]
       i_2 \leftarrow i_1 + 1
       j_2 \leftarrow j_1 + 4
       t_5 \leftarrow 4 * i_2
       t_6 \leftarrow A + t_5
       t_7 \leftarrow M[t_6]
       if (t_3 > t_7) goto rfalse
       goto loop(i_2, j_2)
```

```
\mathsf{is\_sorted}(A, n) :
       i_0 \leftarrow 0
       j_0 \leftarrow 0
       goto loop(i_0, j_0)
\mathsf{loop}(i_1,j_1):
       t_0 \leftarrow n-1
       if (i_1 \ge t_0) goto rtrue
       t_1 \leftarrow j_1
       t_2 \leftarrow A + t_1
       t_3 \leftarrow M[t_2]
       i_2 \leftarrow i_1 + 1
       j_2 \leftarrow j_1 + 4
       t_5 \leftarrow 4 * i_2
       t_6 \leftarrow A + t_5
       t_7 \leftarrow M[t_6]
       if (t_3 > t_7) goto rfalse
       goto loop(i_2, j_2)
```

Loop hoisting

$$\begin{split} \text{is_sorted}(A,n): \\ i_0 &\leftarrow 0 \\ j_0 \leftarrow 0 \\ \text{goto loop}(i_0,j_0) \\ \text{loop}(i_1,j_1): \\ \hline t_0 &\leftarrow n-1 \\ \text{if } (i_1 \geq t_0) \text{ goto rtrue} \\ t_1 &\leftarrow j_1 \\ t_2 &\leftarrow A+t_1 \\ t_3 &\leftarrow M[t_2] \\ i_2 &\leftarrow i_1+1 \\ j_2 &\leftarrow j_1+4 \\ t_5 &\leftarrow 4*i_2 \\ t_6 &\leftarrow A+t_5 \\ t_7 &\leftarrow M[t_6] \\ \text{if } (t_3 > t_7) \text{ goto rfalse} \\ \text{goto loop}(i_2,j_2) \end{split}$$

$$\begin{array}{l} \operatorname{is_sorted}(A,n): \\ i_0 \leftarrow 0 \\ j_0 \leftarrow 0 \\ \hline t_0 \leftarrow n-1 \\ \operatorname{goto} \operatorname{loop}(i_0,j_0) \\ \operatorname{loop}(i_1,j_1): \\ \operatorname{if}\ (i_1 \geq t_0) \operatorname{goto} \operatorname{rtrue} \\ t_2 \leftarrow A+j_1 \\ t_3 \leftarrow M[t_2] \\ i_2 \leftarrow i_1+1 \\ j_2 \leftarrow j_1+4 \\ t_5 \leftarrow 4*i_2 \\ t_6 \leftarrow A+t_5 \\ t_7 \leftarrow M[t_6] \\ \operatorname{if}\ (t_3 > t_7) \operatorname{goto} \operatorname{rfalse} \\ \operatorname{goto} \operatorname{loop}(i_2,j_2) \\ \end{array}$$
 @ensures $j_0 = 4*i_0$

 $is_sorted(A, n)$: $i_0 \leftarrow 0$ $j_0 \leftarrow 0$ $t_0 \leftarrow n-1$ goto $loop(i_0, j_0)$ $\mathsf{loop}(i_1,j_1)$: if $(i_1 \ge t_0)$ goto rtrue $t_2 \leftarrow A + j_1$ $t_3 \leftarrow M[t_2]$ $i_2 \leftarrow i_1 + 1$ $j_2 \leftarrow j_1 + 4$ $t_5 \leftarrow 4 * i_2$ $t_6 \leftarrow A + t_5$ $t_7 \leftarrow M[t_6]$ if $(t_3 > t_7)$ goto rfalse goto $loop(i_2, j_2)$

 $\mathsf{is_sorted}(A, n) :$ $i_0 \leftarrow 0$ $j_0 \leftarrow 0$ $t_0 \leftarrow n-1$ goto $loop(i_0, j_0)$ $\mathsf{loop}(i_1,j_1)$: if $(i_1 \ge t_0)$ goto rtrue $t_2 \leftarrow A + j_1$ $t_3 \leftarrow M[t_2]$ $i_2 \leftarrow i_1 + 1$ $j_2 \leftarrow j_1 + 4$ $t_6 \leftarrow A + j_2$ $t_7 \leftarrow M[t_6]$ if $(t_3 > t_7)$ goto rfalse goto $loop(i_2, j_2)$

@ensures $j_0 = 4 * i_0$

 $@ requires j_1 = 4 * i_1$

 $@ensures j_2 = 4 * i_2$

```
\mathsf{is\_sorted}(A, n) :
       i_0 \leftarrow 0
       j_0 \leftarrow 0
       t_0 \leftarrow n-1
       goto loop(i_0, j_0)
\mathsf{loop}(i_1, j_1):
       if (i_1 \ge t_0) goto rtrue
       t_2 \leftarrow A + j_1
       t_3 \leftarrow M[t_2]
       i_2 \leftarrow i_1 + 1
       j_2 \leftarrow j_1 + 4
       t_6 \leftarrow A + j_2
       t_7 \leftarrow M[t_6]
       if (t_3 > t_7) goto rfalse
       goto loop(i_2, j_2)
```

t2 is another induction variable.

```
\mathsf{is\_sorted}(A, n) :
       i_0 \leftarrow 0
       j_0 \leftarrow 0
       t_0 \leftarrow n-1
       goto loop(i_0, j_0)
\mathsf{loop}(i_1, j_1):
       if (i_1 \ge t_0) goto rtrue
       t_2 \leftarrow A + j_1
       t_3 \leftarrow M[t_2]
       i_2 \leftarrow i_1 + 1
       j_2 \leftarrow j_1 + 4
       t_6 \leftarrow A + j_2
       t_7 \leftarrow M[t_6]
       if (t_3 > t_7) goto rfalse
       goto loop(i_2, j_2)
```

t2 is another induction variable.

We introduce k to keep track of k = A + j

$$\mathsf{is_sorted}(A, n)$$
 :

$$i_0 \leftarrow 0$$

$$j_0 \leftarrow 0$$

$$k_0 \leftarrow A + j_0$$

$$t_0 \leftarrow n-1$$

goto $loop(i_0, j_0, k_0)$

$$loop(i_1, j_1, k_1) :$$

if $(i_1 \ge t_0)$ goto rtrue

$$t_2 \leftarrow k_1$$

$$t_3 \leftarrow M[t_2]$$

$$i_2 \leftarrow i_1 + 1$$

$$j_2 \leftarrow j_1 + 4$$

$$k_2 \leftarrow k_1 + 4$$

$$t_6 \leftarrow A + j_2$$

$$t_7 \leftarrow M[t_6]$$

if $(t_3 > t_7)$ goto rfalse goto loop (i_2, j_2, k_2)

$$@ensures j_0 = 4 * i_0$$

@ensures
$$k_0 = A + j_0$$

@requires
$$j_1 = 4 * i_1 \wedge k_1 = A + j_1$$

@ensures
$$j_2 = 4 * i_2$$

$$@ensures k_2 = A + j_2$$

$$k_2 = A + j_2 = A + j_1 + 4 = k_1 + 4$$

```
is\_sorted(A, n):
       i_0 \leftarrow 0
       j_0 \leftarrow 0
       k_0 \leftarrow A + j_0
       t_0 \leftarrow n-1
       goto loop(i_0, j_0, k_0)
loop(i_1, j_1, k_1) :
       if (i_1 \ge t_0) goto rtrue
       t_2 \leftarrow k_1
       t_3 \leftarrow M[t_2]
       i_2 \leftarrow i_1 + 1
       j_2 \leftarrow j_1 + 4
       k_2 \leftarrow k_1 + 4
       t_6 \leftarrow A + j_2
       t_7 \leftarrow M[t_6]
       if (t_3 > t_7) goto rfalse
       goto loop(i_2, j_2, k_2)
```

$$\begin{split} \text{is_sorted}(A,n): \\ i_0 &\leftarrow 0 \\ j_0 \leftarrow 0 \\ k_0 \leftarrow A \\ t_0 \leftarrow n-1 \\ \text{goto loop}(i_0,j_0,k_0) \\ \text{loop}(i_1,j_1,k_1): \\ \text{if } (i_1 \geq t_0) \text{ goto rtrue} \\ t_3 \leftarrow M[k_1] \\ i_2 \leftarrow i_1 + 1 \\ j_2 \leftarrow j_1 + 4 \\ k_2 \leftarrow k_1 + 4 \\ t_7 \leftarrow M[k_2] \\ \text{if } (t_3 > t_7) \text{ goto rfalse} \\ \text{goto loop}(i_2,j_2,k_2) \end{split}$$

@ensures
$$j_0 = 4*i_0$$
@ensures $k_0 = A+j_0$
@requires $j_1 = 4*i_1 \wedge$

@ensures $j_2 = 4 * i_2$

@ensures $k_2 = A + j_2$

```
is\_sorted(A, n):
       i_0 \leftarrow 0
       j_0 \leftarrow 0
       k_0 \leftarrow A + j_0
       t_0 \leftarrow n-1
       goto loop(i_0, j_0, k_0)
loop(i_1, j_1, k_1) :
       if (i_1 \ge t_0) goto rtrue
       t_2 \leftarrow k_1
       t_3 \leftarrow M[t_2]
       i_2 \leftarrow i_1 + 1
       j_2 \leftarrow j_1 + 4
       k_2 \leftarrow k_1 + 4
       t_6 \leftarrow A + j_2
       t_7 \leftarrow M[t_6]
       if (t_3 > t_7) goto rfalse
       goto loop(i_2, j_2, k_2)
```

One more round of constant propagation, dead code elim., and CSE

 $is_sorted(A, n)$: $i_0 \leftarrow 0$ $j_0 \leftarrow 0$ $k_0 \leftarrow A + j_0$ $t_0 \leftarrow n-1$ goto $loop(i_0, j_0, k_0)$ $loop(i_1, j_1, k_1) :$ if $(i_1 \ge t_0)$ goto rtrue $t_2 \leftarrow k_1$ $t_3 \leftarrow M[t_2]$ $i_2 \leftarrow i_1 + 1$ $j_2 \leftarrow j_1 + 4$ $k_2 \leftarrow k_1 + 4$ $t_6 \leftarrow A + j_2$ $t_7 \leftarrow M[t_6]$ if $(t_3 > t_7)$ goto rfalse goto $loop(i_2, j_2, k_2)$

$$\begin{array}{ll} \operatorname{is_sorted}(A,n): \\ i_0 \leftarrow 0 \\ j_0 \leftarrow 0 \\ k_0 \leftarrow A \\ 0 \leftarrow n-1 \\ \operatorname{goto} \operatorname{loop}(i_0,j_0,k_0) \\ \operatorname{loop}(i_1,j_1,k_1): \\ \operatorname{if}\ (i_1 \geq t_0) \operatorname{goto} \operatorname{rtrue} \\ t_3 \leftarrow M[k_1] \\ i_2 \leftarrow i_1+1 \\ j_2 \leftarrow j_1+4 \\ k_2 \leftarrow k_1+4 \\ t_7 \leftarrow M[k_2] \\ \operatorname{if}\ (t_3 > t_7) \operatorname{goto} \operatorname{rfalse} \\ \end{array} \right. \\ \operatorname{@ensures}\ j_0 = 4*i_0 \\ \operatorname{@ensures}\ k_0 = A+j_0 \\ \operatorname{@ensures}\ j_1 = 4*i_1 \land 1 \\ \operatorname{@ensures}\ j_2 = 4*i_2 \\ \operatorname{@ensures}\ k_2 = A+j_2 \\ \operatorname{@ensu$$

One more round of constant propagation, dead code elim., and CSE

goto $loop(i_2, j_2, k_2)$

$$\begin{split} \text{is_sorted}(A,n): \\ i_0 &\leftarrow 0 \\ k_0 &\leftarrow A + j_0 \\ t_0 &\leftarrow n-1 \\ \text{goto loop}(i_0,j_0,k_0) \\ \text{loop}(i_1,j_1,k_1): \\ \text{if } (i_1 \geq t_0) \text{ goto rtrue} \\ t_2 &\leftarrow k_1 \\ t_3 &\leftarrow M[t_2] \\ i_2 &\leftarrow i_1 + 1 \\ j_2 &\leftarrow j_1 + 4 \\ k_2 &\leftarrow k_1 + 4 \\ t_6 &\leftarrow A + j_2 \\ t_7 &\leftarrow M[t_6] \\ \text{if } (t_3 > t_7) \text{ goto rfalse} \\ \text{goto loop}(i_2,j_2,k_2) \end{split}$$

$$\begin{array}{ll} \text{is_sorted}(A,n): \\ i_0 \leftarrow 0 \\ j_0 \leftarrow 0 \\ k_0 \leftarrow A \\ t_0 \leftarrow n-1 \\ \text{goto loop}(i_0,j_0,k_0) \\ \\ \text{loop}(i_1,j_1,k_1): \\ \text{if } (i_1 \geq t_0) \text{ goto rtrue} \\ t_3 \leftarrow M[k_1] \\ i_2 \leftarrow i_1+1 \\ j_2 \leftarrow j_1+4 \\ k_2 \leftarrow k_1+4 \\ t_7 \leftarrow M[k_2] \\ \text{if } (t_3 > t_7) \text{ goto rfalse} \\ \text{goto loop}(i_2,j_2,k_2) \\ \end{array}$$

One more round of constant propagation, dead code elim., and CSE

$$\begin{split} \text{is_sorted}(A,n): \\ i_0 &\leftarrow 0 \\ k_0 &\leftarrow A + j_0 \\ t_0 &\leftarrow n-1 \\ \text{goto loop}(i_0,j_0,k_0) \\ \text{loop}(i_1,j_1,k_1): \\ \text{if } (i_1 \geq t_0) \text{ goto rtrue} \\ t_2 &\leftarrow k_1 \\ t_3 &\leftarrow M[t_2] \\ i_2 &\leftarrow i_1 + 1 \\ j_2 &\leftarrow j_1 + 4 \\ k_2 &\leftarrow k_1 + 4 \\ t_6 &\leftarrow A + j_2 \\ t_7 &\leftarrow M[t_6] \\ \text{if } (t_3 > t_7) \text{ goto rfalse} \\ \text{goto loop}(i_2,j_2,k_2) \end{split}$$

$$\begin{array}{lll} \operatorname{is_sorted}(A,n): \\ i_0 \leftarrow 0 \\ j_0 \leftarrow 0 \\ k_0 \leftarrow A \\ t_0 \leftarrow n-1 \\ \operatorname{goto} \operatorname{loop}(i_0,j_0,k_0) \\ \operatorname{loop}(i_1,j_1,k_1): \\ \operatorname{if}\ (i_1 \geq t_0) \operatorname{goto} \operatorname{rtrue} \\ t_3 \leftarrow M[k_1] \\ i_2 \leftarrow i_1+1 \\ j_2 \leftarrow j_1+4 \\ k_2 \leftarrow k_1+4 \\ k_2 \leftarrow k_1+4 \\ t_7 \leftarrow M[k_2] \\ \operatorname{if}\ (t_3 > t_7) \operatorname{goto} \operatorname{rfalse} \\ \operatorname{goto} \operatorname{loop}(i_2,j_2,k_2) \end{array}$$
 @ensures $j_0 = 4*i_0$ @ensures $j_1 = 4*i_1 \land k_1 \land k_2 \land k_3 \land k_4 \land k_4 \land k_5 \land k_5$

One more round of constant propagation, dead code elim., and CSE

```
\mathsf{is\_sorted}(A, n) :
       i_0 \leftarrow 0
       j_0 \leftarrow 0
       k_0 \leftarrow A
       t_0 \leftarrow n-1
       goto loop(i_0, j_0, k_0)
loop(i_1, j_1, k_1) :
       if (i_1 \ge t_0) goto rtrue
       t_3 \leftarrow M[k_1]
       i_2 \leftarrow i_1 + 1
       j_2 \leftarrow j_1 + 4
       k_2 \leftarrow k_1 + 4
       t_7 \leftarrow M[k_2]
       if (t_3 > t_7) goto rfalse
       goto loop(i_2, j_2, k_2)
```

```
\mathsf{is\_sorted}(A, n) :
       i_0 \leftarrow 0
       j_0 \leftarrow 0
       k_0 \leftarrow A
       t_0 \leftarrow n-1
       goto loop(i_0, j_0, k_0)
loop(i_1, j_1, k_1) :
       if (i_1 \ge t_0) goto rtrue
       t_3 \leftarrow M[k_1]
       i_2 \leftarrow i_1 + 1
       j_2 \leftarrow j_1 + 4
       k_2 \leftarrow k_1 + 4
       t_7 \leftarrow M[k_2]
       if (t_3 > t_7) goto rfalse
       goto loop(i_2, j_2, k_2)
```

j1, j2, and j3 are no longer needed.

```
\mathsf{is\_sorted}(A, n) :
       i_0 \leftarrow 0
       j_0 \leftarrow 0
       k_0 \leftarrow A
       t_0 \leftarrow n-1
       goto loop(i_0, j_0, k_0)
loop(i_1, j_1, k_1) :
       if (i_1 \ge t_0) goto rtrue
       t_3 \leftarrow M[k_1]
       i_2 \leftarrow i_1 + 1
       j_2 \leftarrow j_1 + 4
       k_2 \leftarrow k_1 + 4
       t_7 \leftarrow M[k_2]
       if (t_3 > t_7) goto rfalse
       goto loop(i_2, j_2, k_2)
```

j1, j2, and j3 are no longer needed.

 $is_sorted(A, n)$: $i_0 \leftarrow 0$ $j_0 \leftarrow 0$ $k_0 \leftarrow A$ $t_0 \leftarrow n-1$ goto $loop(i_0, j_0, k_0)$ $loop(i_1, j_1, k_1) :$ if $(i_1 \ge t_0)$ goto rtrue $t_3 \leftarrow M[k_1]$ $i_2 \leftarrow i_1 + 1$ $j_2 \leftarrow j_1 + 4$ $k_2 \leftarrow k_1 + 4$ $t_7 \leftarrow M[k_2]$ if $(t_3 > t_7)$ goto rfalse goto $loop(i_2, j_2, k_2)$

j1, j2, and j3 are no longer needed.

 $is_sorted(A, n)$: $i_0 \leftarrow 0$ $k_0 \leftarrow A$ $t_0 \leftarrow n-1$ goto $loop(i_0, k_0)$ $\mathsf{loop}(i_1, k_1)$: if $(i_1 \ge t_0)$ goto rtrue $t_3 \leftarrow M[k_1]$ $i_2 \leftarrow i_1 + 1$ $k_2 \leftarrow k_1 + 4$ $t_7 \leftarrow M[k_2]$ if $(t_3 > t_7)$ goto rfalse

goto $loop(i_2, k_2)$

 $@ensures k_0 = A + 4 * i_0$

@requires $k_1 = A + 4 * i_1$

@ensures $k_2 = A + 4 * i_2$

```
\mathsf{is\_sorted}(A, n) :
       i_0 \leftarrow 0
       k_0 \leftarrow A
      t_0 \leftarrow n-1
      goto loop(i_0, k_0)
loop(i_1, k_1):
       if (i_1 \ge t_0) goto rtrue
       t_3 \leftarrow M[k_1]
      i_2 \leftarrow i_1 + 1
       k_2 \leftarrow k_1 + 4
      t_7 \leftarrow M[k_2]
       if (t_3 > t_7) goto rfalse
       goto loop(i_2, k_2)
```

```
\mathsf{is\_sorted}(A, n) :
       i_0 \leftarrow 0
       k_0 \leftarrow A
       t_0 \leftarrow n-1
       goto loop(i_0, k_0)
\mathsf{loop}(i_1, k_1):
       if (i_1 \ge t_0) goto rtrue
       t_3 \leftarrow M[k_1]
       i_2 \leftarrow i_1 + 1
       k_2 \leftarrow k_1 + 4
       t_7 \leftarrow M[k_2]
       if (t_3 > t_7) goto rfalse
       goto loop(i_2, k_2)
```

$i_1 \ge t_0 \text{ iff } A + 4 * i_1 \ge A + 4 * t_0$

```
is\_sorted(A, n):
       i_0 \leftarrow 0
       k_0 \leftarrow A
       t_0 \leftarrow n-1
       goto loop(i_0, k_0)
\mathsf{loop}(i_1, k_1):
       if (i_1 \ge t_0) goto rtrue
       t_3 \leftarrow M[k_1]
       i_2 \leftarrow i_1 + 1
       k_2 \leftarrow k_1 + 4
       t_7 \leftarrow M[k_2]
       if (t_3 > t_7) goto rfalse
       goto loop(i_2, k_2)
```

$$\begin{split} \text{is_sorted}(A,n): \\ i_0 &\leftarrow 0 \\ k_0 \leftarrow A \\ t_0 \leftarrow n-1 \\ \text{goto loop}(i_0,k_0) \\ \text{loop}(i_1,k_1): \\ \text{if } (k_1 \geq A+4*t_0) \text{ goto rtrue} \\ t_3 \leftarrow M[k_1] \\ i_2 \leftarrow i_1+1 \\ k_2 \leftarrow k_1+4 \\ t_7 \leftarrow M[k_2] \\ \text{if } (t_3 > t_7) \text{ goto rfalse} \\ \text{goto loop}(i_2,k_2) \end{split}$$

$i_1 \ge t_0 \text{ iff } A + 4 * i_1 \ge A + 4 * t_0$

```
\mathsf{is\_sorted}(A, n) :
       i_0 \leftarrow 0
       k_0 \leftarrow A
       t_0 \leftarrow n-1
       goto loop(i_0, k_0)
\mathsf{loop}(i_1, k_1):
       if (k_1 \ge A + 4 * t_0) goto rtrue
       t_3 \leftarrow M[k_1]
       i_2 \leftarrow i_1 + 1
       k_2 \leftarrow k_1 + 4
       t_7 \leftarrow M[k_2]
       if (t_3 > t_7) goto rfalse
       goto loop(i_2, k_2)
```

```
\mathsf{is\_sorted}(A, n) :
      i_0 \leftarrow 0
       k_0 \leftarrow A
       t_0 \leftarrow n-1
       goto loop(i_0, k_0)
\mathsf{loop}(i_1, k_1):
       if (k_1 \ge A + 4 * t_0) goto rtrue
       t_3 \leftarrow M[k_1]
       i_2 \leftarrow i_1 + 1
       k_2 \leftarrow k_1 + 4
       t_7 \leftarrow M[k_2]
       if (t_3 > t_7) goto rfalse
       goto loop(i_2, k_2)
```

i0, i1, and i2 are no longer needed.

 $is_sorted(A, n)$: $i_0 \leftarrow 0$ $k_0 \leftarrow A$ $t_0 \leftarrow n-1$ goto $loop(i_0, k_0)$ $\mathsf{loop}(i_1, k_1)$: if $(k_1 \ge A + 4 * t_0)$ goto rtrue $t_3 \leftarrow M[k_1]$ $i_2 \leftarrow i_1 + 1$ $k_2 \leftarrow k_1 + 4$ $t_7 \leftarrow M[k_2]$ if $(t_3 > t_7)$ goto rfalse goto $loop(i_2, k_2)$

i0, i1, and i2 are no longer needed.

$$\begin{array}{c} \operatorname{is_sorted}(A,n): \\ k_0 \leftarrow A \\ t_0 \leftarrow n-1 \\ t_8 \leftarrow 4*t_0 \\ t_9 \leftarrow A+t_8 \\ \operatorname{goto} \operatorname{loop}(k_0) \\ \operatorname{loop}(k_1): \\ \operatorname{if}\ (k_1 \geq t_9) \operatorname{goto} \operatorname{rtrue} \\ t_3 \leftarrow M[k_1] \\ k_2 \leftarrow k_1+4 \\ t_7 \leftarrow M[k_2] \\ \operatorname{if}\ (t_3 > t_7) \operatorname{goto} \operatorname{rfalse} \\ \operatorname{goto} \operatorname{loop}(k_2) \\ \end{array}$$

```
is\_sorted(A, n):
      i_0 \leftarrow 0
      k_0 \leftarrow A
      t_0 \leftarrow n-1
      goto loop(i_0, k_0)
\mathsf{loop}(i_1, k_1):
       if (k_1 \ge A + 4 * t_0) goto rtrue
      t_3 \leftarrow M[k_1]
      i_2 \leftarrow i_1 + 1
      k_2 \leftarrow k_1 + 4
      t_7 \leftarrow M[k_2]
       if (t_3 > t_7) goto rfalse
      goto loop(i_2, k_2)
```

i0, i1, and i2 are no longer needed.

$$\begin{aligned} \text{is_sorted}(A,n): \\ k_0 &\leftarrow A \\ t_0 &\leftarrow n-1 \\ t_8 &\leftarrow 4*t_0 \\ t_9 &\leftarrow A+t_8 \\ \text{goto loop}(k_0) \\ \text{loop}(k_1): \\ \text{if } (k_1 \geq t_9) \text{ goto rtrue} \\ t_3 &\leftarrow M[k_1] \\ k_2 &\leftarrow k_1+4 \\ t_7 &\leftarrow M[k_2] \\ \text{if } (t_3 > t_7) \text{ goto rfalse} \\ \text{goto loop}(k_2) \end{aligned}$$

Unrolling the loop once can remove one memory access per iteration.