Programming Paradigms

Control Flow (Part 5)

Prof. Dr. Michael Pradel

Software Lab, University of Stuttgart

Summer 2020

Overview

- Expression Evaluation
- Structured and Unstructured Control Flow
- Selection
- Iteration
- Recursion 

Recursion

- **Equally powerful as iteration**
- **Most PLs allow both recursion and iteration**
  - Iteration: More natural in imperative PLs (because the loop body typically updates variables)
  - Recursion: More natural in functional PLs (because the recursive function typically doesn't update any non-local variables)

Efficiency

Naively written or naively compiled recursive functions: Less efficient than equivalent iterative code

- Reason: New allocation frame for each call
- Example: Compute $\sum_{low \le i \le high} f(i)$ in Scheme

```
(define sum
  (lambda (f low high)
    (if (= low high)
      (f low)
      (+ (f low) (sum f (+ low 1) high)))))
```

Efficiency

Naively written or naively compiled recursive functions: Less efficient than equivalent iterative code

- Reason: New allocation frame for each call
- Example: Compute $\sum_{low \le i \le high} f(i)$ in Scheme

```
(define sum
  (lambda (f low high)
    (if (= low high)
      (f low)
      (+ (f low) (sum f (+ low 1) high)))))
```

Then and else branches

Efficiency

Naively written or naively compiled recursive functions: Less efficient than equivalent iterative code

- Reason: New allocation frame for each call
- Example: Compute $\sum_{low \le i \le high} f(i)$ in Scheme

```
(define sum
  (lambda (f low high)
    (if (= low high)
      (f low)
      (+ (f low) (sum f (+ low 1) high)))))
```

Tail Recursion

Recursive call is the last statement before the function returns

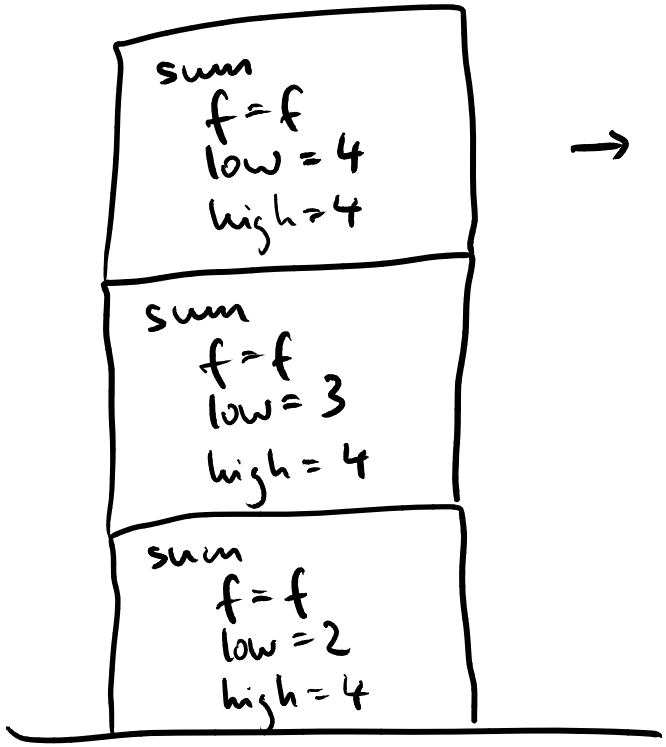
- Compiled code can reuse same allocation frame
- Revised example:

```
(define sum
  (lambda (f low high subtotal)
    (if (= low high)
      (+ subtotal (f low))
      (sum f (+ low 1) high (+ subtotal (f low))))))
```

Example: Summation

(sum f 2 4)

Naive implementation

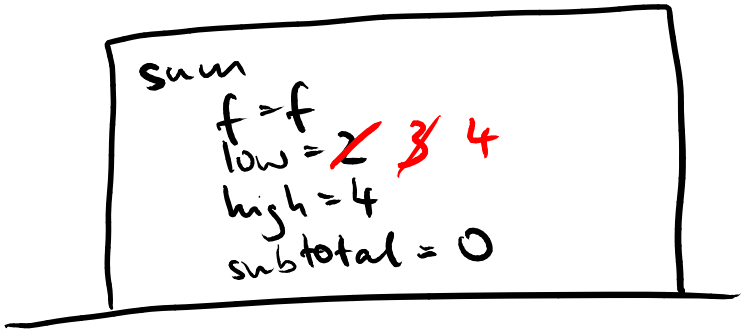


→ 3 allocation frames

(sum f 2 4 0)

Tail-recursive implementation

→ reusing a single allocation frame



Overview

- Expression Evaluation
- Structured and Unstructured Control Flow
- Selection
- Iteration
- Recursion ✔