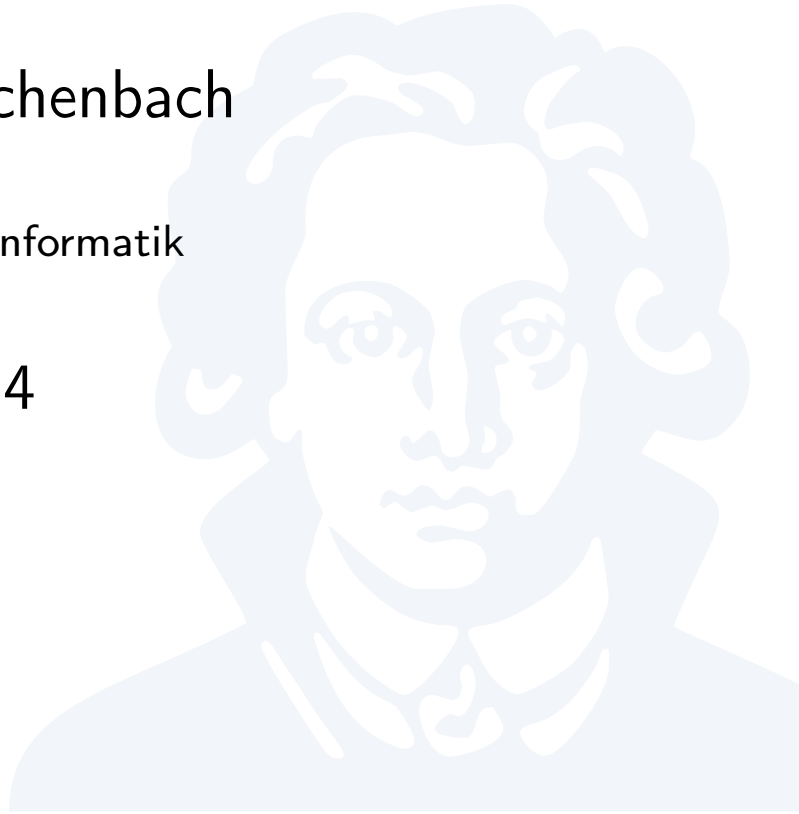
Foundations of Programming Languages

Nested Subprograms

Prof. Dr. Christoph Reichenbach

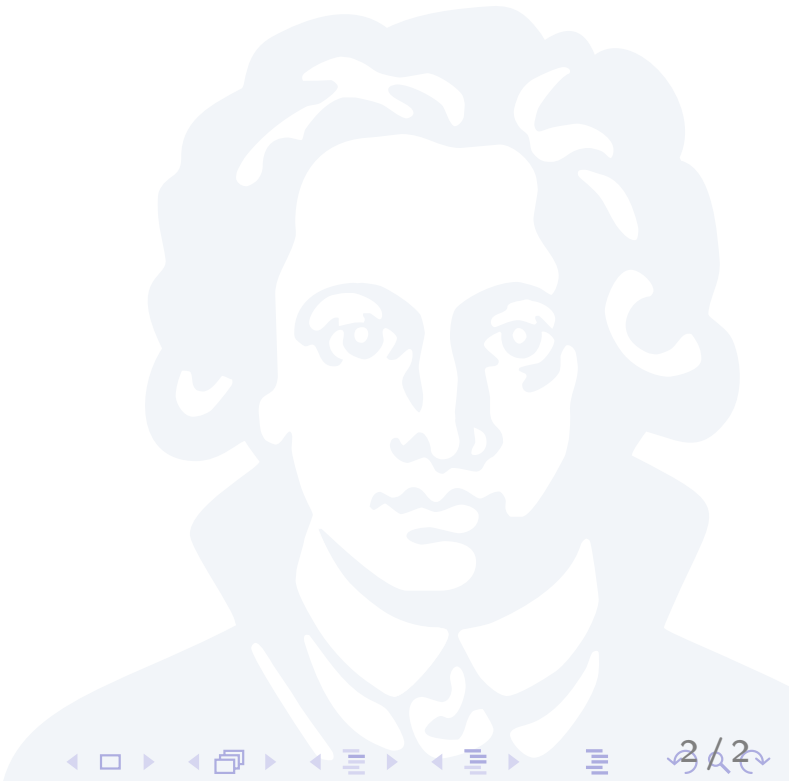
Fachbereich 12 / Institut für Informatik

29. Oktober 2014



Nested Subprograms

```
subprogram f()
var x := 1;
  subprogram g()
  begin
     print(x);
  end
begin
   return g;
end
```



Nested Subprograms

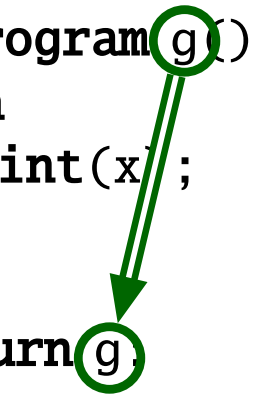
```
subprogram f()
var x := 1;
  subprogram g()
  begin
     print(x);
  end
begin
   return g;
end

subprogram h(p)
var x := 2;
begin
   p();
end
```

Nested Subprograms

```
subprogram f()
var x := 1;
  subprogram g()
  begin
     print(x);
  end
begin
   return g;
end

subprogram h(p)
var x := 2;
begin
   p();
end

var x := 3;
h(f());
```

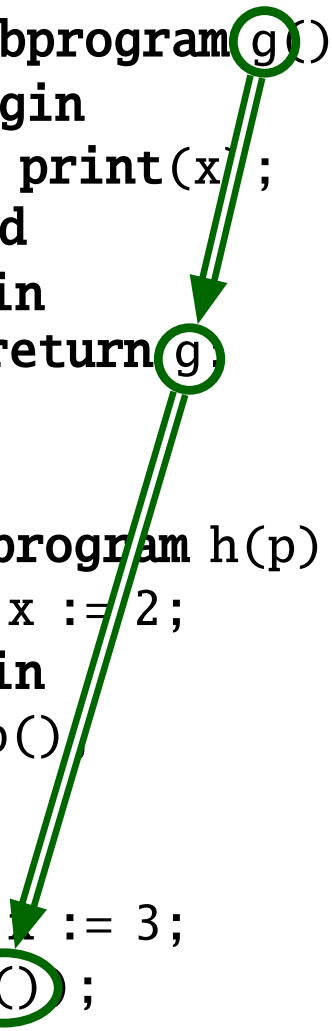
Nested Subprograms

```
subprogram f()
var x := 1;
  subprogram g()
  begin
     print(x);
  end
begin
   return g;
end

subprogram h(p)
var x := 2;
begin
   p();
end

var x := 3;
h(f());
```



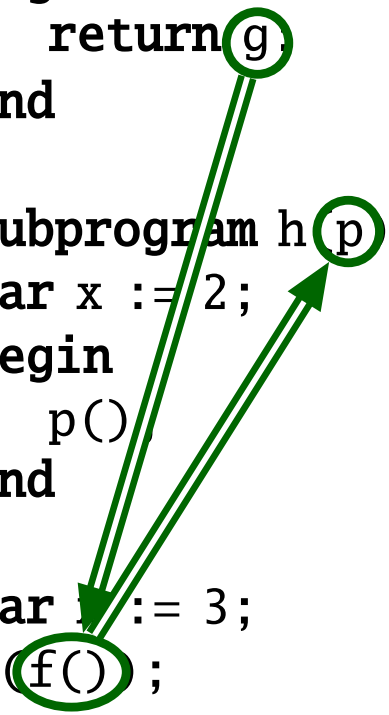
Nested Subprograms

```
subprogram f()
var x := 1;
  subprogram g()
  begin
     print(x);
  end
begin
   return g;
end

subprogram h(p)
var x := 2;
begin
   p()
end

var f := 3;
h(f());
```



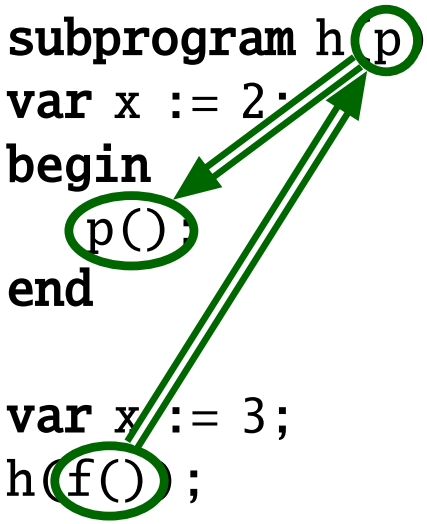
Nested Subprograms

```
subprogram f()
var x := 1;
  subprogram g()
  begin
     print(x);
  end
begin
   return g;
end

subprogram h(p)
var x := 2;
begin
   p()
end

var f := 3;
h(f());
```



Nested Subprograms

```
subprogram f()
var x := 1;
  subprogram g()
  begin
     print(x);
  end
begin
   return g;
end

subprogram h(p)
var x := 2;
begin
   p();
end

var x := 3;
h(f());
```



Nested Subprograms

```
subprogram f()
var x := 1;
  subprogram g()
  begin
     print(x);
  end
begin
   return g;
end

subprogram h(p)
var x := 2;
begin
   p();
end

var x := 3;
h(f());
```

Nested Subprograms

```
subprogram f()
var x := 1;
  subprogram g()
  begin
     print(x);
  end
begin
   return g;
end

subprogram h(p)
var x := 2;
begin
   p();
end

var x := 3;
h(f());
```

Nested Subprograms

```
subprogram f()
var x := 1;
  subprogram g()
  begin
     print(x);
  end
begin
   return g;
end

subprogram h(p)
var x := 2;
begin
   p();
end

var x := 3;
h(f());
```

- *deep binding*:
  - subprograms use static scope
  - print 1

Nested Subprograms

```
subprogram f()
var x := 1;
  subprogram g()
  begin
     print(x);
  end
begin
   return g;
end

subprogram h(p)
var x := 2;
begin
   p();
end

var x := 3;
h(f());
```

- *deep binding*:
  - subprograms use static scope
  - print 1
- *shallow binding*:
  - subprograms use dynamic scope
  - print 2

Nested Subprograms

```
subprogram f()
var x := 1;
  subprogram g()
  begin
     print(x);
  end
begin
   return g;
end

subprogram h(p)
var x := 2;
begin
   p();
end

var x := 3;
h(f());
```

- *deep binding*:
  - subprograms use static scope
  - print 1
- *shallow binding*:
  - subprograms use dynamic scope
  - print 2
- *ad-hoc binding*:
  - subprograms acquire scope from call site that binds subprogram parameter
  - print 3