

Problem 0

Workspace

Stack

Heap

```
let y = 3 in
let f (z : int) : int =
  y * z * z in
let x = f 5 in
x
```

Problem 0

Workspace

Stack

Heap

```
let y = 3 in  
let f (z : int) : int =  
    y * z * z in  
let x = f 5 in  
x
```

Problem 0

Workspace

```
let f (z : int) : int =  
    y * z * z in  
let x = f 5 in  
x
```

Stack

y	3
---	---

Heap

Problem 0

Workspace

```
let f (z : int) : int =  
      y * z * z in  
let x = f 5 in  
x
```

Stack

y	3
---	---

Heap

Problem 0

Workspace

```
let f : int->int =  
  fun (z: int) -> y * z * z in  
let x = f 5 in  
x
```

Stack

y	3
---	---

Heap

Problem 0

Workspace

```
let f : int->int =  
fun (z: int) -> y * z * z in  
let x = f 5 in  
x
```

Stack


y	3
---	---

Heap

Problem 0

Workspace

```
let f : int ->int =      in  
let x = f 5 in  
x
```



Stack

y	3
---	---

Heap

```
fun (z: int) -> y * z * z
```

Problem 0

Workspace


```
let f : int ->int =   in  
let x = f 5 in  
x
```

Stack

y	3
---	---

Heap

```
fun (z: int) -> y * z * z
```



Problem 0

Workspace

```
let x = f 5 in  
x
```

Stack

y	3
---	---

f	
---	---

Heap

fun (z: int) -> y * z * z


Problem 0

Workspace

```
let x = f 5 in  
x
```

Stack

y	3
---	---

f	
---	---


Heap

fun (z: int) -> y * z * z

Problem 0

Workspace

```
let x = 5 in  
x
```



Stack

y	3
---	---

f	
---	---

Heap

```
fun (z: int) -> y * z * z
```

Problem 0

Workspace

```
let x = 5 in  
x
```

Stack

y	3
---	---

f	•
---	---

Heap

fun (z: int) -> y * z * z

Function
call!

Problem 0

Workspace

`y * z * z`

Stack

y	3
---	---

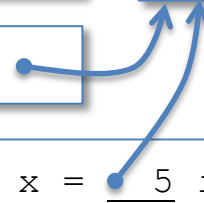
f	•
---	---

let x = <u>5</u> in
x

z	5
---	---

Heap

fun (z: int) -> y * z * z



Problem 0

Workspace

y * z * z

Stack

y	3
---	---

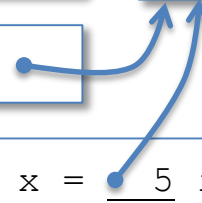
f	•
---	---

let x = <u>5</u> in x

z	5
---	---

Heap

fun (z: int) -> y * z * z



Problem 0

Workspace

3 * z * z

Stack

y | 3

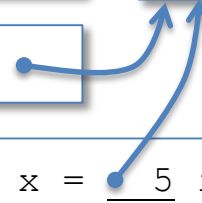
f | •

let x = 5 in
x

z | 5

Heap

fun (z: int) -> y * z * z



Problem 0

Workspace

3 * z * z

Stack

y	3
---	---

f	•
---	---

let x = <u>5</u> in x

z	5
---	---

Heap

fun (z: int) -> y * z * z



Problem 0

Workspace

3 * 5 * z

Stack

y	3
---	---

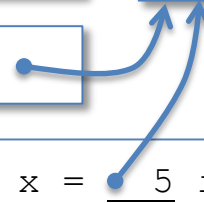
f	•
---	---

let x = <u>5</u> in x

z	5
---	---

Heap

fun (z: int) -> y * z * z



Problem 0

Workspace

3 * 5 * z

Stack

y	3
---	---

f	•
---	---

let x = <u>5</u> in x

z	5
---	---

Heap

fun (z: int) -> y * z * z



Problem 0

Workspace

15 * z

Stack

y	3
---	---

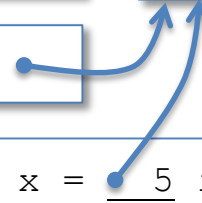
f	•
---	---

let x = <u>5</u> in
x

z	5
---	---

Heap

fun (z: int) -> y * z * z



Problem 0

Workspace

15 * z

Stack

y	3
---	---

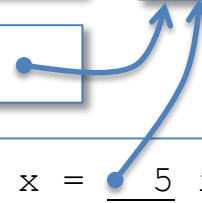
f	•
---	---

let x = <u>5</u> in x

z	5
---	---

Heap

fun (z: int) -> y * z * z



Problem 0

Workspace

15 * 5

Stack

y	3
---	---

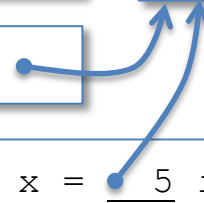
f	•
---	---

let x = <u>5</u> in
x

z	5
---	---

Heap

fun (z: int) -> y * z * z



Problem 0

Workspace

15 * 5

Stack

y	3
---	---

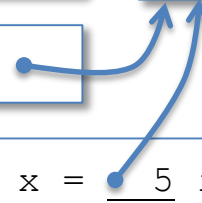
f	•
---	---

let x = <u>5</u> in
x

z	5
---	---

Heap

fun (z: int) -> y * z * z



Problem 0

Workspace

75

Stack

y	3
---	---

f	•
---	---

let x = <u>5</u> in
x

z	5
---	---

Heap

fun (z: int) -> y * z * z

Function
return!

Problem 0

Workspace

```
let x = 75 in  
x
```

Stack

y	3
---	---

f	
---	---

Heap

fun (z: int) -> y * z * z

Problem 0

Workspace

```
let x = 75 in  
x
```

Stack

y	3
---	---

f	
---	---

Heap

fun (z: int) -> y * z * z



Problem 0

Workspace

x

Stack

y 22

f

x 75

Heap

fun (z: int) -> y * z * z



Answer


Problem 0

Workspace

x

Stack

y	22
---	----

f	
---	---

x	75
---	----

Heap

fun (z: int) -> y * z * z


Problem 0

Workspace

75

Stack

y	22
---	----

f	
---	---

x	75
---	----

Heap

fun (z: int) -> y * z * z



DONE!