

# 1 sum\_to\_n\_rec — MIR Walkthrough

**Purpose:** TODO: Describe why this walkthrough exists

## 1.1 Source Context

```
fn sum_to_n_rec(n:u32) -> u32 {
    if n == 0 {
        0
    } else {
        n + sum_to_n_rec(n - 1)
    }
}
```

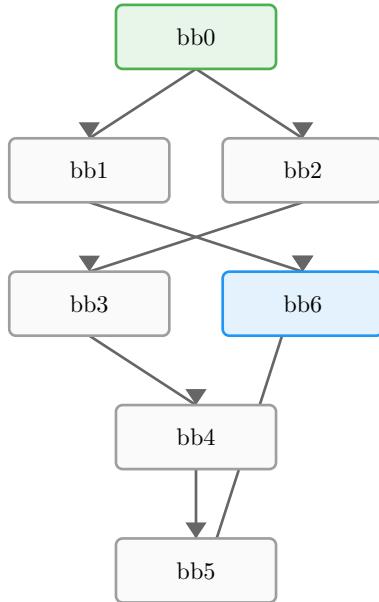
## 1.2 Function Overview

- **Function:** sum\_to\_n\_rec
- **Basic blocks:** 7
- **Return type:** u32
- **Notable properties:**
  - Contains panic path
  - Uses checked arithmetic
  - Recursive
  - Contains assertions
  - Has conditional branches

## 1.3 Locals

Local	Type	Notes
0	u32	Return place
1	u32	
2	u32	
3	u32	
4	(u32, bool)	
5	(u32, bool)	

## 1.4 Control-Flow Overview



## 1.5 Basic Blocks

### 1.5.1 bb0 — entry

*Entry point of the function.*

MIR	Annotation
<code>→ switch(\_1) \[0-&gt;bb1; else-&gt;bb2\]</code>	Branch on <code>\_1</code>

### 1.5.2 bb1

MIR	Annotation
<code>\_0 = 0</code>	Load constant
<code>→ goto bb6</code>	Jump to bb6

### 1.5.3 bb2

MIR	Annotation
<code>\_4 = checked(\_1 - 1)</code>	Checked Subtract (may panic)
<code>→ assert(move \_4.1 == false) → bb3</code>	Panic if move <code>\_4.1</code> is true

### 1.5.4 bb3

MIR	Annotation
<code>\_3 = move \_4.0</code>	Move value
<code>→ \_2 = sum\_to\_n\_rec(move \_3) → bb4</code>	Recursive call to <code>sum_to_n_rec</code>

### 1.5.5 bb4

MIR	Annotation
<code>\_5 = checked(\_1 + \_2)</code>	Checked Add (may panic)
<code>→ assert(move \_5.1 == false) → bb5</code>	Panic if move <code>\_5.1</code> is true

### 1.5.6 bb5

MIR	Annotation
\_0 = move \_5.0	Move value
→ goto bb6	Jump to bb6

### 1.5.7 bb6 — return / success

*Normal return path.*

MIR	Annotation
→ return	Return from function

## 1.6 Key Observations

TODO: Add bullet points summarizing what this MIR teaches

- 
- 

## 1.7 Takeaways

TODO: One or two sentences to generalize this example

## 2 main — MIR Walkthrough

**Purpose:** TODO: Describe why this walkthrough exists

### 2.1 Source Context

```
fn main() {
    let ans = sum_to_n_rec(10);

    assert!(ans == 55);
}
```

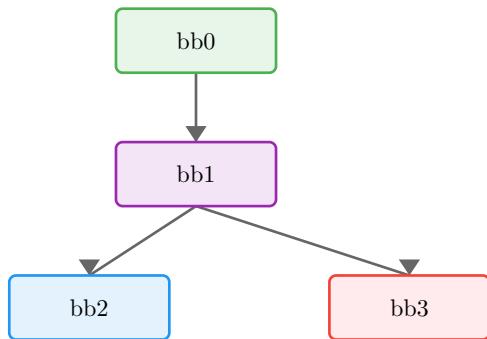
### 2.2 Function Overview

- **Function:** main
- **Basic blocks:** 4
- **Return type:** ()
- **Notable properties:**
  - Contains panic path
  - Has conditional branches

### 2.3 Locals

Local	Type	Notes
0	()	Return place
1	u32	
2	!	

### 2.4 Control-Flow Overview



### 2.5 Basic Blocks

#### 2.5.1 bb0 — entry

*Entry point of the function.*

MIR	Annotation
$\rightarrow \_1 = \text{sum\_to\_n\_rec}(10) \rightarrow \text{bb1}$	Call sum_to_n_rec

#### 2.5.2 bb1 — branch point

MIR	Annotation
$\rightarrow \text{switch}(\_1) \ \backslash [55 \rightarrow \text{bb2}; \text{else} \rightarrow \text{bb3}]$	Branch on $\_1$

### **2.5.3 bb2** — return / success

*Normal return path.*

MIR	Annotation
→ <code>return</code>	Return from function

### **2.5.4 bb3** — panic path

*Panic/diverging path.*

MIR	Annotation
→ <code>\_2 = panic(\[16 bytes\])</code>	Call panic

## 2.6 Key Observations

TODO: Add bullet points summarizing what this MIR teaches

- 
- 

## 2.7 Takeaways

TODO: One or two sentences to generalize this example

