

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)  
    }  
}
```

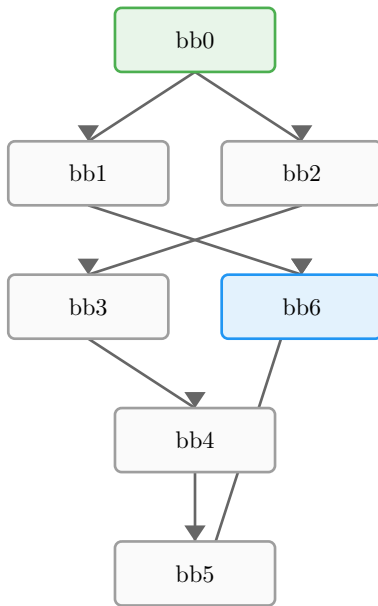
1.2 Function Overview

- **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
→ switch(_1) [0→bb1; else→bb2]	Branch on _1

1.5.2 bb1

MIR	Annotation
_0 = 0	Load constant
→ goto bb6	Jump to bb6

1.5.3 bb2

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

1.5.4 bb3

MIR	Annotation
_3 = move _4.0	Move value
→ _2 = sum_to_n_rec(move _3) → bb4	Recursive call to sum_to_n_rec

1.5.5 bb4

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

1.5.6 bb5

MIR	Annotation
<code>_0 = move _5.0</code>	Move value
<code>→ goto bb6</code>	Jump to bb6

1.5.7 bb6 — return / success

Normal return path.

MIR	Annotation
<code>→ return</code>	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);  
}
```

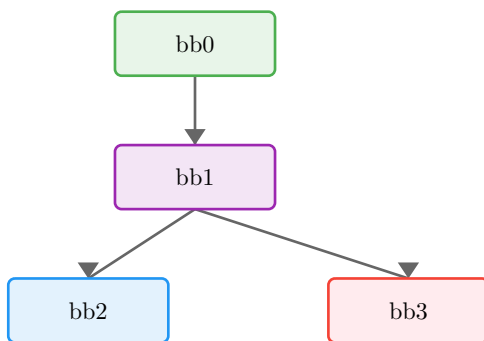
2.2 Function Overview

- **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
→ <code>_1 = sum_to_n_rec(10)</code> → bb1	Call <code>sum_to_n_rec</code>

2.5.2 bb1 — branch point

MIR	Annotation
→ <code>switch(_1) [55→bb2; else→bb3]</code>	Branch on <code>_1</code>

2.5.3 bb2 — return / success

Normal return path.

MIR	Annotation
→ return	Return from function

2.5.4 bb3 — panic path

Panic/diverging path.

MIR	Annotation
→ _2 = panic([16 bytes])	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

