```
1   pub fn add(a: i32, b: i32) -> i32 {
2       a + b
3   }
```

```
1       pub fn add(a: i32, b: i32) -> i32 {
```

```
3       }
4
5       pub fn sub(a: i32, b: i32) -> i32 {
6   -       a - b
6   +       let c = a - b;
7   +       c
8       }
9
10      pub fn mul(a: i32, b: i32) -> i32 {
```

```
15          a / b
16      }
```

```
(
  to_show_lines: (
    (
      tag: "Equal",
      old_index: 0,
      new_index: 0,
      content: "pub fn add(a: i32, b: i32) -> i32 {\n",
    ),
    (
      tag: "Equal",
      old_index: 1,
      new_index: 1,
      content: "    a + b\n",
    ),
    (
      tag: "Equal",
      old_index: 2,
      new_index: 2,
      content: "}\n",
    ),
    (
      tag: "Equal",
      old_index: 3,
      new_index: 3,
      content: "\n",
    ),
    (
      tag: "Equal",
```

```
      old_index: 4,
      new_index: 4,
      content: "pub fn sub(a: i32, b: i32) -> i32 {\n",
    ),
    (
      tag: "Delete",
      old_index: 5,
      new_index: none,
      content: "    a - b\n",
    ),
    (
      tag: "Insert",
      old_index: none,
      new_index: 5,
      content: "    let c = a - b;\n",
    ),
    (
      tag: "Insert",
      old_index: none,
      new_index: 6,
      content: "    c\n",
    ),
    (
      tag: "Equal",
      old_index: 6,
      new_index: 7,
      content: "}\n",
    ),
    (
      tag: "Equal",
      old_index: 7,
      new_index: 8,
      content: "\n",
    ),
    (
      tag: "Equal",
      old_index: 8,
      new_index: 9,
      content: "pub fn mul(a: i32, b: i32) -> i32 {\n",
    ),
    (
      tag: "Spacer",
      old_index: none,
      new_index: none,
      content: "",
    ),
    (
      tag: "Equal",
      old_index: 14,
      new_index: 15,
      content: "}\n",
    ),
  ),
  lines: (
    (
      tag: "Equal",
```

```
    old_index: 0,
    new_index: 0,
    content: "pub fn add(a: i32, b: i32) -> i32 {\n",
),
(
    tag: "Equal",
    old_index: 1,
    new_index: 1,
    content: "    a + b\n",
),
(
    tag: "Equal",
    old_index: 2,
    new_index: 2,
    content: "}\n",
),
(
    tag: "Equal",
    old_index: 3,
    new_index: 3,
    content: "\n",
),
(
    tag: "Equal",
    old_index: 4,
    new_index: 4,
    content: "pub fn sub(a: i32, b: i32) -> i32 {\n",
),
(
    tag: "Delete",
    old_index: 5,
    new_index: none,
    content: "    a - b\n",
),
(
    tag: "Insert",
    old_index: none,
    new_index: 5,
    content: "    let c = a - b;\n",
),
(
    tag: "Insert",
    old_index: none,
    new_index: 6,
    content: "    c\n",
),
(
    tag: "Equal",
    old_index: 6,
    new_index: 7,
    content: "}\n",
),
(
    tag: "Equal",
    old_index: 7,
    new_index: 8,
```

```
            content: "\n",
        ),
        (
            tag: "Equal",
            old_index: 8,
            new_index: 9,
            content: "pub fn mul(a: i32, b: i32) -> i32 {\n",
        ),
        (
            tag: "Equal",
            old_index: 9,
            new_index: 10,
            content: "    a * b\n",
        ),
        (
            tag: "Equal",
            old_index: 10,
            new_index: 11,
            content: "}\n",
        ),
        (
            tag: "Equal",
            old_index: 11,
            new_index: 12,
            content: "\n",
        ),
        (
            tag: "Equal",
            old_index: 12,
            new_index: 13,
            content: "pub fn div(a: i32, b: i32) -> i32 {\n",
        ),
        (
            tag: "Equal",
            old_index: 13,
            new_index: 14,
            content: "    a / b\n",
        ),
        (
            tag: "Equal",
            old_index: 14,
            new_index: 15,
            content: "}\n",
        ),
    ),
)
```