

Source Program Function

```
// uniquePaths algorithm
func uniquePaths(m int, n
int) int {
  memo := make([][]int, m)
  for i := 0; i < m; i++ {
    ...
}
func main(){...}</pre>
```

Source Program Cleaned

```
func uniquePaths(m int, n
int) int {
memo := make([][]int, m)
for i := 0; i < m; i++ {
...
}</pre>
```

LLM Generated Rust Program

```
fn uniquePaths(m: i32, n:
i32) -> i32 {
let mut memo :=
vec![vec![1; n as usize];
m as usize];
for i in 1..m{
...
}
```

Error Messages

```
error[E0308]: mismatched
types-->
rust_programs/src/uniquep
aths.rs:11:14
11 | map.insert(num,
index as i32);
| ----- ^^^ expected
`i32`, found &i32
```

Test Harness

```
kani::check!().with_type:
:<(i32,i32)>().cloned().f
or_each(|(m, n)|{
let result = fn
unique_paths_llm(i32,
i32);
let result_prime = fn
unique_paths_rwasm(m, n);
assert_eq!(result,
result_prime);
```