```
; preds = %for.inc, %entry
for.cond:
                 \%0 = \text{load i} 32, i 32*\% i, align 4
                  %cmp = icmp ult i32 %0, 100
          br i1 %cmp, label %land.rhs, label %land.end
                                       ; preds = %for.cond
    land.rhs:
%1 = load %struct.list_node*, %struct.list_node** %list_it, align 8
         %cmp1 = icmp ne %struct.list_node* %1, null
                       br label %land.end
```

```
invoke.cont: ; preds = %for.body
%4 = load %struct.list_node*, %struct.list_node** %list_it, align 8
%next = getelementptr inbounds %struct.list_node, %struct.list_node* %4, i32 0, i32 1
store %struct.list_node* %3, %struct.list_node** %next, align 8
%5 = load i32, i32* %i, align 4
%6 = load %struct.list_node*, %struct.list_node** %list_it, align 8
%value = getelementptr inbounds %struct.list_node, %struct.list_node* %6, i32 0, i32 0
store i32 %5, i32* %value, align 8
br label %for.inc
```

land.end:

[10/11]

%2 = phi i1 [ false, %for.cond ], [ %cmp1, %land.rhs ]

br i1 %2, label %for.body, label %for.end

; preds = %land.rhs, %for.cond

```
[8/9]

for.body: ; preds = %land.end

%call = call i8* @_Znwm(i64 16) #7

%3 = bitcast i8* %call to %struct.list_node*

call void @_ZN9list_nodeC2Ev(%struct.list_node* %3)

br label %invoke.cont
```

```
for.inc: ; preds = %invoke.cont
%7 = load %struct.list_node*, %struct.list_node** %list_it, align 8
%next2 = getelementptr inbounds %struct.list_node, %struct.list_node* %7, i32 0, i32 1
%8 = load %struct.list_node*, %struct.list_node** %next2, align 8
store %struct.list_node* %8, %struct.list_node** %list_it, align 8
%9 = load i32, i32* %i, align 4
%inc = add i32 %9, 1
store i32 %inc, i32* %i, align 4
br label %for.cond
```