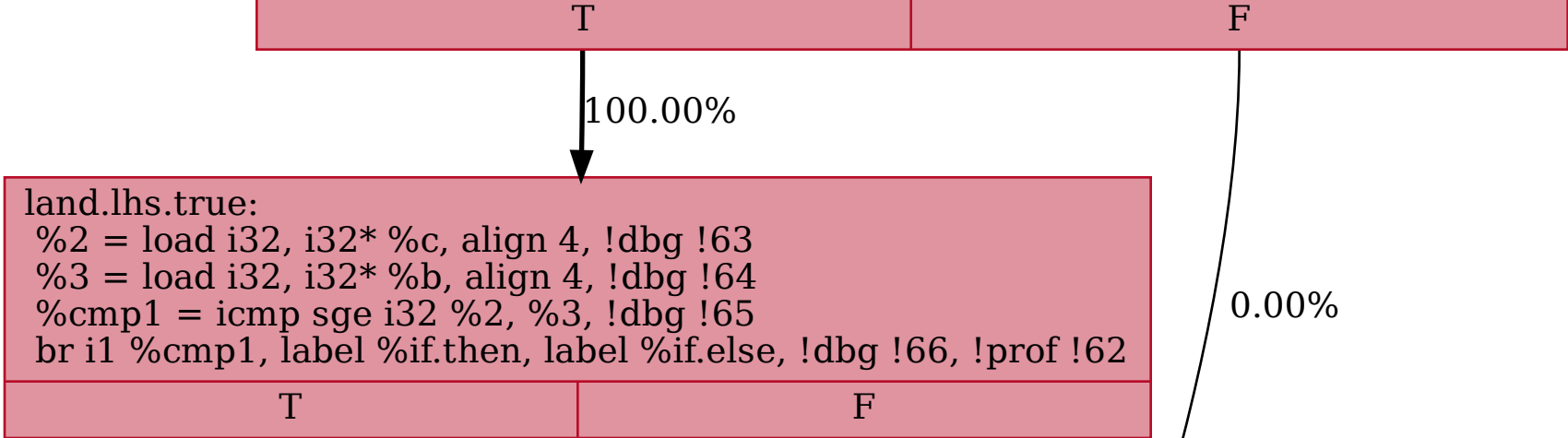


```
entry:
  %retval = alloca i32, align 4
  %a = alloca i32, align 4
  %b = alloca i32, align 4
  %c = alloca i32, align 4
  %d = alloca i32, align 4
  %x = alloca i32, align 4
  %y = alloca i32, align 4
  %z = alloca i32, align 4
  store i32 0, i32* %retval, align 4
  call void @llvm.dbg.declare(metadata i32* %a, metadata !43, metadata
... !DIExpression()), !dbg !44
  store i32 0, i32* %a, align 4, !dbg !44
  call void @llvm.dbg.declare(metadata i32* %b, metadata !45, metadata
... !DIExpression()), !dbg !46
  store i32 -1, i32* %b, align 4, !dbg !46
  call void @llvm.dbg.declare(metadata i32* %c, metadata !47, metadata
... !DIExpression()), !dbg !48
  store i32 10, i32* %c, align 4, !dbg !48
  call void @llvm.dbg.declare(metadata i32* %d, metadata !49, metadata
... !DIExpression()), !dbg !50
  store i32 11, i32* %d, align 4, !dbg !50
  call void @llvm.dbg.declare(metadata i32* %x, metadata !51, metadata
... !DIExpression()), !dbg !52
  store i32 12, i32* %x, align 4, !dbg !52
  call void @llvm.dbg.declare(metadata i32* %y, metadata !53, metadata
... !DIExpression()), !dbg !54
  store i32 0, i32* %y, align 4, !dbg !54
  call void @llvm.dbg.declare(metadata i32* %z, metadata !55, metadata
... !DIExpression()), !dbg !56
  store i32 4, i32* %z, align 4, !dbg !56
  %0 = load i32, i32* %a, align 4, !dbg !57
  %1 = load i32, i32* %d, align 4, !dbg !59
  %cmp = icmp slt i32 %0, %1, !dbg !60
  br i1 %cmp, label %land.lhs.true, label %if.else, !dbg !61, !prof !62
```



```
if.then:
  %4 = load i32, i32* %x, align 4, !dbg !67
  store i32 %4, i32* %y, align 4, !dbg !69
  br label %if.end, !dbg !70
```

```
if.else:
  %5 = load i32, i32* %x, align 4, !dbg !71
  %add = add nsw i32 %5, 1, !dbg !73
  store i32 %add, i32* %y, align 4, !dbg !74
  br label %if.end
```

```
if.end:
  %6 = load i32, i32* %a, align 4, !dbg !75
  %7 = load i32, i32* %d, align 4, !dbg !77
  %cmp2 = icmp slt i32 %6, %7, !dbg !78
  %conv = zext i1 %cmp2 to i32, !dbg !79
  %8 = load i32, i32* %c, align 4, !dbg !80
  %9 = load i32, i32* %b, align 4, !dbg !81
  %cmp3 = icmp sge i32 %8, %9, !dbg !82
  %conv4 = zext i1 %cmp3 to i32, !dbg !83
  %and = and i32 %conv, %conv4, !dbg !84
  %tobool = icmp ne i32 %and, 0, !dbg !79
  br i1 %tobool, label %if.then5, label %if.else6, !dbg !85, !prof !62
```

```
if.then5:
  %10 = load i32, i32* %x, align 4, !dbg !86
  store i32 %10, i32* %y, align 4, !dbg !88
  br label %if.end8, !dbg !89
```

```
if.else6:
  %11 = load i32, i32* %x, align 4, !dbg !90
  %add7 = add nsw i32 %11, 1, !dbg !92
  store i32 %add7, i32* %y, align 4, !dbg !93
  br label %if.end8
```

```
if.end8:
  %12 = load i32, i32* %a, align 4, !dbg !94
  %13 = load i32, i32* %d, align 4, !dbg !96
  %cmp9 = icmp slt i32 %12, %13, !dbg !97
  %conv10 = zext i1 %cmp9 to i32, !dbg !98
  %14 = load i32, i32* %c, align 4, !dbg !99
  %15 = load i32, i32* %b, align 4, !dbg !100
  %cmp11 = icmp sge i32 %14, %15, !dbg !101
  %conv12 = zext i1 %cmp11 to i32, !dbg !102
  %or = or i32 %conv10, %conv12, !dbg !103
  %tobool13 = icmp ne i32 %or, 0, !dbg !98
  br i1 %tobool13, label %if.then14, label %if.else15, !dbg !104, !prof !62
```

```
if.then14:
  %16 = load i32, i32* %x, align 4, !dbg !105
  store i32 %16, i32* %y, align 4, !dbg !107
  br label %if.end17, !dbg !108
```

```
if.else15:
  %17 = load i32, i32* %x, align 4, !dbg !109
  %add16 = add nsw i32 %17, 1, !dbg !111
  store i32 %add16, i32* %y, align 4, !dbg !112
  br label %if.end17
```

```
if.end17:
  ret i32 0, !dbg !113
```

CFG for 'main' function