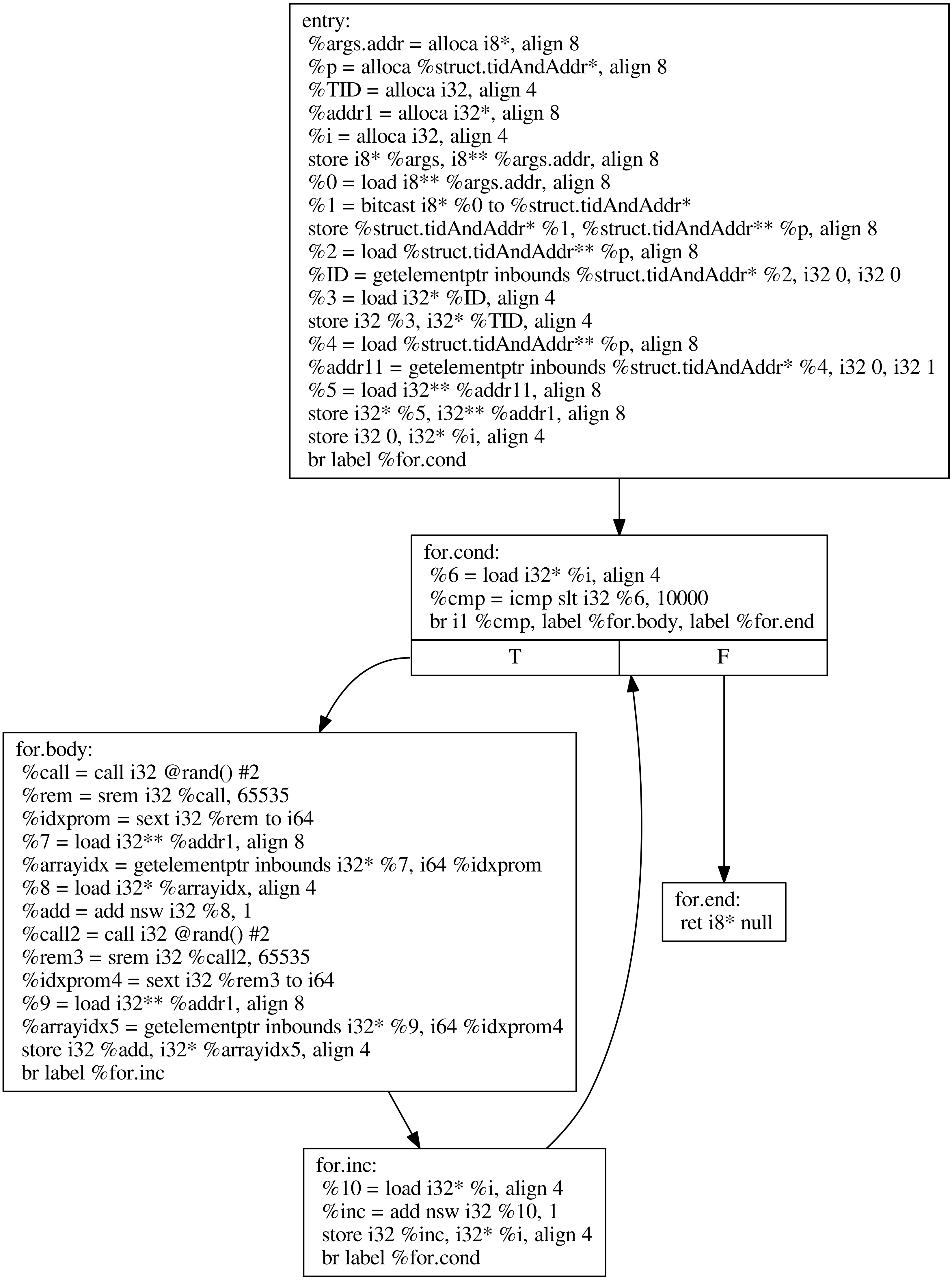

```
entry:
 %args.addr = alloca i8*, align 8
 %p = alloca %struct.tidAndAddr*, align 8
 %TID = alloca i32, align 4
 %addr1 = alloca i32*, align 8
 %i = alloca i32, align 4
 store i8* %args, i8** %args.addr, align 8
 %0 = load i8** %args.addr, align 8
 %1 = bitcast i8* %0 to %struct.tidAndAddr*
 store %struct.tidAndAddr* %1, %struct.tidAndAddr** %p, align 8
 %2 = load %struct.tidAndAddr** %p, align 8
 %ID = getelementptr inbounds %struct.tidAndAddr* %2, i32 0, i32 0
 %3 = load i32* %ID, align 4
 store i32 %3, i32* %TID, align 4
 %4 = load %struct.tidAndAddr** %p, align 8
 %addr11 = getelementptr inbounds %struct.tidAndAddr* %4, i32 0, i32 1
 %5 = load i32** %addr11, align 8
 store i32* %5, i32** %addr1, align 8
 store i32 0, i32* %i, align 4
 br label %for.cond
```

```
for.cond:
 %6 = load i32* %i, align 4
 %cmp = icmp slt i32 %6, 10000
 br i1 %cmp, label %for.body, label %for.end
```

```
for.body:
 %call = call i32 @rand() #2
 %rem = srem i32 %call, 65535
 %idxprom = sext i32 %rem to i64
 %7 = load i32** %addr1, align 8
 %arrayidx = getelementptr inbounds i32* %7, i64 %idxprom
 %8 = load i32* %arrayidx, align 4
 %add = add nsw i32 %8, 1
 %call2 = call i32 @rand() #2
 %rem3 = srem i32 %call2, 65535
 %idxprom4 = sext i32 %rem3 to i64
 %9 = load i32** %addr1, align 8
 %arrayidx5 = getelementptr inbounds i32* %9, i64 %idxprom4
 store i32 %add, i32* %arrayidx5, align 4
 br label %for.inc
```

```
for.end:
 ret i8* null
```

```
for.inc:
 %10 = load i32* %i, align 4
 %inc = add nsw i32 %10, 1
 store i32 %inc, i32* %i, align 4
 br label %for.cond
```

CFG for '_Z6DoWorkPv' function