

```

entry:
  %a = alloca i32, align 4
  %b = alloca i32, align 4
  %c = alloca i32, align 4
  %d = alloca i32, align 4
  %e = alloca i32, align 4
  %f = alloca i32, align 4
  store i32 0, i32* %a, align 4
  store i32 1, i32* %c, align 4
  br label %do.body

```

```

do.body:
  %0 = load i32, i32* %a, align 4
  %add = add nsw i32 %0, 1
  store i32 %add, i32* %c, align 4
  %1 = load i32, i32* %c, align 4
  %2 = load i32, i32* %b, align 4
  %mul = mul nsw i32 %1, %2
  store i32 %mul, i32* %c, align 4
  %3 = load i32, i32* %b, align 4
  %cmp = icmp sgt i32 %3, 9
  br i1 %cmp, label %if.then, label %if.else

```

T

F

```

if.then:
  %4 = load i32, i32* %d, align 4
  %5 = load i32, i32* %c, align 4
  %mul1 = mul nsw i32 %4, %5
  store i32 %mul1, i32* %f, align 4
  %6 = load i32, i32* %f, align 4
  %sub = sub nsw i32 %6, 3
  store i32 %sub, i32* %c, align 4
  br label %if.end

```

```

if.else:
  %7 = load i32, i32* %e, align 4
  %add2 = add nsw i32 %7, 1
  store i32 %add2, i32* %a, align 4
  %8 = load i32, i32* %d, align 4
  %div = sdiv i32 %8, 2
  store i32 %div, i32* %e, align 4
  br label %if.end

```

```

if.end:
  %9 = load i32, i32* %b, align 4
  store i32 %9, i32* %a, align 4
  br label %do.cond

```

```

do.cond:
  %10 = load i32, i32* %a, align 4
  %cmp3 = icmp slt i32 %10, 9
  br i1 %cmp3, label %do.body, label %do.end

```

T

F

```

do.end:
  %11 = load i32, i32* %a, align 4
  %add4 = add nsw i32 %11, 1
  store i32 %add4, i32* %a, align 4
  ret void

```

CFG for 'test' function