```
entry:
 %m = alloca i32, align 4
 %call = call ...
 %i = load i32, i32* %m, align 4
 switch i32 %i, label %sw.default [
 i32 2, label %sw.bb
 i32 4, label %sw.bb1
 i32 6, label %sw.bb4
 ]
```

def	2	4	6

```
sw.default:
 br label %sw.epilog
```

```
sw.bb:
 %mul = mul nsw i32 2, 1
 %add = add nsw i32 %mul, 5
 br label %sw.epilog
```

```
sw.bb1:
 %mul2 = mul nsw i32 2, 1
 %add3 = add nsw i32 %mul2, 5
 br label %sw.epilog
```

```
sw.bb4:
 %mul5 = mul nsw i32 2, 1
 %add6 = add nsw i32 %mul5, 1
 br label %sw.epilog
```

```
sw.epilog:
 %x.0 = phi i32 [ 2, %sw.default ], [ %add6, %sw.bb4 ], [ %add3, %sw.bb1 ], [
 ... %add, %sw.bb ]
 %tau = call i32 (...) @llvm.tau.i32(i32 %x.0,
 i32 %add, i32 %add3, i32 %add6)
 %mul7 = mul nsw i32 2, %tau
 %add8 = add nsw i32 %mul7, 10
 %add9 = add nsw i32 9, %tau
 %cmp = icmp sle i32 %add8, %add9
 br i1 %cmp, label %if.end, label %if.else
```

T	F

```
if.else:
 %tau1 = call i32 (...) @llvm.tau.i32(i32 %tau,
 i32 %add3, i32 %add6)
 %mul10 = mul nsw i32 3, %tau1
 %add11 = add nsw i32 %add8, %mul10
 switch i32 %add11, label %if.end [
 i32 200, label %sw.bb13
 i32 300, label %sw.bb15
 ]
```

def	200	300

```
if.end:
 %tau3 = call i32 (...) @llvm.tau.i32(i32 %tau,
 i32 %add, i32 %add3)
 %add17 = add nsw i32 %add8, %tau3
 store i32 %add17, i32* %m, align 4
 br label %end
```

```
sw.bb13:
 %call14 = call ...
 br label %end
```

```
sw.bb15:
 call void @exit(i32 0) #7
 unreachable
```

```
end:
 %add18 = add nsw i32 1, %tau
 ret i32 0
```

Program in HPSSA Form