```
Path P_0 Path P_1 Path P_2
                             entry:
                              %m = alloca i32, align 4
                             %call = call nonnull align 8 dereferenceable(16)
                             ... %"class.std::basic istream"* @ ZNSirsERi(%"class.std::basic istream"* nonnull
                             ... align 8 dereferenceable(16) @ \(\overline{\bar{Z}}\)St3cin, i\(\overline{\bar{3}}\)2* nonnull\(\overline{\bar{a}}\)align 4
                                 dereferenceable(4) %m)
                             %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
                                                             2
                                      def
                        sw.default:
                                                 sw.bb:
                                                                          sw.bb1:
                                                                                                    sw.bb4:
                                                                           br label\%sw.epilog
                                                                                                    br label %sw.epilog
                         br label %sw.epilog
                                                  br label %sw.epilog
                                sw.epilog:
                                %n.0 = phi i32 [ undef, %sw.default ], [ 5, %sw.bb4\], [ 5, %sw.bb1 ], [/10,
                                %x.0 = phi i32 [ 2, %sw.default ], [ 3, %sw.bb4 ], [\7, %sw.bb1 ], [ 7,
                                ... %sw.bb ]
                                %tau4 = call i32 (...) @llvm.tau.i32(i32 %x.0, i32 7, i32 7, i32 3)
                                 %tau = call i32 (...) @llvm.tau.i32(i32 %n.0, i32 5, i32 5, i32 10)
                                 %mul8 = mul nsw/i32 2, %tau4
                                 %add9 = add nsw i32 %mul8, 10
                                 %add10 = add nsw i32 9, %tau4
                                 %cmp = icmp s te i32 %add9, %add10
                                 br i1 %cmp, Vabel %if.end, label %if.else
                                                                                                                   \%tau1 Speculative - \
                                                                                                             ConstantRange < 5, 6 > 1
                                                             if.else:
                                                             %tau5 = call i32 (...) @llvm.tau.i32(i32 %tau4, i32 7, i32 3) %tau1 = call i32 (...) @llvm.tau.i32(i32 %tau, i32 5, i32 5)
                                                              %mul11 = mul nsw i32 3, %tau5
                                                              %add12 = add nsw i32 %tau1, %mul11
                                                              switch i32 %add12, label %if.end [
                                                              j32 200, label %sw.bb14
                                                             <u>i</u>32 300, label %sw.bb15
  1.80 \% tau7 Speculative
                                                                     def
                                                                                           200
                                                                                                                  300
ConstantRange < 7, 8 >
                 if.end:
                  %tau7 = call i32 (...) @llvm.tau.i32(i32\%tau4, i32 7, i32 7)
                                                                                                         sw.bb15:
                  %tau3 = call i32 (...) @llvm.tau.i32(i32 %tau, i32 5, i32 10)
                                                                                      sw.ibb14:
                                                                                                          call void @exit(i32 0) #7
                  %add17 = add nsw i32 %add9, %tau7
                                                                                       br label %end
                                                                                                          unreachable
                  store i32 %add17, i32* %m, align 4
                  br label %end
                                                                  end:
                                                                  ret i32,0
                                                          Speculative SCCP on 'main' function
```