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) @  $\overline{Z}$ St3cin, i32\* nonnull 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 def 2 6 sw.bb1: sw.bb4: sw.bb: %mul5 = mul nsw i32 2, 1 %mul2 = mul nsw i32 2, 1 sw.default: %mul = mul nsw i32 2, 1 %add3 = add nsw i32 %mul2, 5 %add6 = add nsw i32 %mul5, 1 br label %sw.epilog %add = add nsw i32 %mul, 5 %sub = sub nsw i32 %add3, 2 %add7 = add nsw i32 %add6, 2 br label %sw.epilog br label %sw.epilog br label %sw.epilog sw.epilog: %n.0 = phi i32 [ undef, %sw.default ], [ %add7, %sw.bb4 ], [ %sub, %sw.bb1 ...], [ 10, %sw.bb ] %x.0 = phi i32 [ 2, %sw.default ], [ %add6, %sw.bb4 ], [ %add3, %sw.bb1 ], [ ... %add, %sw.bb ] %tau4 = call i32 (...) @llvm.tau.i32(i32 %x.0, i32 %add, i32 %add3, i32 ... %add6) %tau = call i32 (...) @llvm.tau.i32(i32 %n.0, i32 %sub, i32 %add7, i32 10) %mul8 = mul nsw i32 2, %tau4 %add9 = add nsw i32 %mul8, 10 %add10 = add nsw i32 9, %tau4 %cmp = icmp sle i32 %add9, %add10 br i1 %cmp, label %if.end, label %if.else if.else: %tau5 = call i32 (...) @llvm.tau.i32(i32 %tau4, i32 %add3, i32 %add6) %tau1 = call i32 (...) @llvm.tau.i32(i32 %tau, i32 %sub, i32 %add7) %mul11 = mul nsw i32 3, %tau5 %add12 = add nsw i32 %tau1, %mul11 switch i32 %add12, label %if.end [ i32 200, label %sw.bb14 i32 300, label %sw.bb15 def 200 300 if.end: %tau7 = call i32 (...) @llvm.tau.i32(i32 %tau4, i32 %add, i32 %add3) sw.bb15: %tau3 = call i32 (...) @llvm.tau.i32(i32 %tau, i32 %sub, i32 10) sw.bb14: 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 Program in HPSSA Form