br i1 %cmp, label %for.body, label %for.end, !dbg !1299 %inc = add nsw i32 %i.0, 1, !dbg !1314 %0 = load i32, i32\* %n, align 4, !dbg !1297 br label %for.cond, !dbg !1315, !llvm.loop !1316 br label %for.inc, !dbg !1313 %i.0 = phi i32 [ 0, %entry ], [ %inc, %for.inc ], !dbg !1295 %index = alloca [50 x i32], align 16 %idxprom5 = sext i32 %i.0 to i64, !dbg !1309 %skip = alloca [50 x i32], align 16 %idxprom = sext i32 %i.0 to i64, !dbg !1300 %cmp = icmp slt i32 %i.0, %0, !dbg !1298 %idxprom7 = sext i32 %i.0 to i64, !dbg !1310 %arrayidx6 = getelementptr inbounds [50 x i32], [50 x i32]\* %index, i64 0, i64 %idxprom5, !dbg !1309 %arrayidx = getelementptr inbounds [50 x i32], [50 x i32]\* %skip, i64 0, i64 %idxprom, !dbg !1300 %arrayidx8 = getelementptr inbounds [50 x i32], [50 x i32]\* %index, i64 0, i64 %idxprom7, !dbg !1310 %1 = load i32, i32\* %arrayidx, align 4, !dbg !1300 %4 = load i32, i32\* %arrayidx8, align 4, !dbg !1310 %3 = load i32, i32\* %arrayidx6, align 4, !dbg !1309 %b = alloca [100 x i32], align 16 %idxprom9 = sext i32 %4 to i64, !dbg !1311 %tobool = icmp ne i32 %1, 0, !dbg !1300 br i1 %tobool, label %if.end, label %if.then, !dbg !1303 %arrayidx10 = getelementptr inbounds [100 x i32], [100 x i32]\* %b, i64 0, i64 %idxprom9, !dbg !1311 %idxprom1 = sext i32 %i.0 to i64, !dbg !1304 store i32 %3, i32\* %arrayidx10, align 4, !dbg !1312 br label %if.end, !dbg !1308 %arrayidx2 = getelementptr inbounds [50 x i32], [50 x i32]\* %index, i64 0, i64 %idxprom1, !dbg !1304 %2 = load i32, i32\* %arrayidx2, align 4, !dbg !1304 %a = alloca [100 x i32], align 16 %idxprom3 = sext i32 %2 to i64, !dbg !1306 %arrayidx4 = getelementptr inbounds [100 x i32], [100 x i32]\* %a, i64 0, i64 %idxprom3, !dbg !1306

ret i32 0, !dbg !1318

br label %for.cond, !dbg !1294

%call = call dereferenceable(280) %class.std::basic\_istream\* @\_ZNSirsERi(%class.std::basic\_istream\* @\_ZSt3cin, i32\* dereferenceable(4) %n), !dbg !12901290

%n = alloca i32, align 4

store i32 0, i32\* %arrayidx4, align 4, !dbg !1307