

%8:  
%9 = sext i32 %3 to i64  
%10 = icmp slt i64 %9, 256  
%11 = select i1 %10, i64 %9, i64 256  
%mul.i.i = shl i64 %5, 8  
%sub.i = add nsw i32 %3, -1  
%mul.i = mul nsw i32 %sub.i, %3  
%12 = icmp ugt i64 %11, 1  
%umax = select i1 %12, i64 %11, i64 1  
%min.iters.check = icmp ult i64 %umax, 8  
br i1 %min.iters.check, label %pre\_\nregion\_for\_entry.entry.i.preheader, label ... %vector.scevcheck

T

F

vector.scevcheck:  
%13 = add nsw i64 %umax, -1  
%14 = add i32 %3, -1  
%15 = mul i32 %14, %3  
%16 = trunc i64 %5 to i32  
%17 = shl i32 %16, 8  
%18 = add nsw i32 %15, %17  
%19 = trunc i64 %13 to i32  
%20 = add i32 %18, %19  
%21 = icmp slt i32 %20, %18  
%22 = icmp ugt i64 %13, 4294967295  
%23 = or i1 %21, %22  
br i1 %23, label %pre\_\nregion\_for\_entry.entry.i.preheader, label %vector.memcheck

T

F

vector.memcheck:  
%24 = add i32 %3, -1  
%25 = mul i32 %24, %3  
%26 = sext i32 %25 to i64  
%27 = trunc i64 %5 to i32  
%28 = shl i32 %27, 8  
%29 = sext i32 %28 to i64  
%30 = add nsw i64 %26, %29  
%scevgep = getelementptr float, float\* %2, i64 %30  
%31 = add nsw i64 %umax, %26  
%32 = add nsw i64 %31, %29  
%scevgep2 = getelementptr float, float\* %2, i64 %32  
%scevgep4 = getelementptr float, float\* %1, i64 %30  
%scevgep6 = getelementptr float, float\* %1, i64 %32  
%bound0 = icmp ult float\* %scevgep, %scevgep6  
%bound1 = icmp ult float\* %scevgep4, %scevgep2  
%found.conflict = and i1 %bound0, %bound1  
br i1 %found.conflict, label %pre\_\nregion\_for\_entry.entry.i.preheader, label ... %vector.ph

T

F

vector.ph:  
%n.vec = and i64 %umax, -8  
br label %vector.body

vector.body:  
%index = phi i64 [ 0, %vector.ph ], [ %index.next, %vector.body ]  
%33 = add i64 %index, %mul.i.i  
%34 = trunc i64 %33 to i32  
%35 = add nsw i32 %mul.i, %34  
%36 = sext i32 %35 to i64  
%37 = getelementptr inbounds float, float\* %2, i64 %36  
%38 = bitcast float\* %37 to <8 x float>\*  
%wide.load = load <8 x float>, <8 x float>\* %38, align 4, !tbaa !12, ... !alias.scope !16, !noalias !19  
%39 = getelementptr inbounds float, float\* %1, i64 %36  
%40 = bitcast float\* %39 to <8 x float>\*  
%wide.load8 = load <8 x float>, <8 x float>\* %40, align 4, !tbaa !12, ... !alias.scope !19  
%41 = fdiv <8 x float> %wide.load, %wide.load8, !fpmath !21  
%42 = bitcast float\* %37 to <8 x float>\*  
store <8 x float> %41, <8 x float>\* %42, align 4, !tbaa !12, !alias.scope ... !16, !noalias !19, !llvm.access.group !22  
%index.next = add i64 %index, 8  
%43 = icmp eq i64 %index.next, %n.vec  
br i1 %43, label %middle.block, label %vector.body, !llvm.loop !24

T

F

middle.block:  
%cmp.n = icmp eq i64 %umax, %n.vec  
br i1 %cmp.n, label %adi\_kernel5.exit, label ... %pre\_\nregion\_for\_entry.entry.i.preheader

T

F

pre\_\nregion\_for\_entry.entry.i.preheader:  
%\_local\_id\_x.0.ph = phi i64 [ 0, %vector.memcheck ], [ 0, %vector.scevcheck ... ], [ 0, %8 ], [ %n.vec, %middle.block ]  
br label %pre\_\nregion\_for\_entry.entry.i

pre\_\nregion\_for\_entry.entry.i:  
%\_local\_id\_x.0 = phi i64 [ %46, %pre\_\nregion\_for\_entry.entry.i ], [ ... %\_local\_id\_x.0.ph, %pre\_\nregion\_for\_entry.entry.i.preheader ]  
%add1.i.i = add i64 %\_local\_id\_x.0, %mul.i.i  
%conv.i = trunc i64 %add1.i.i to i32  
%add.i = add nsw i32 %mul.i, %conv.i  
%idxprom.i = sext i32 %add.i to i64  
%arrayidx.i = getelementptr inbounds float, float\* %2, i64 %idxprom.i  
%44 = load float, float\* %arrayidx.i, align 4, !tbaa !12  
%arrayidx6.i = getelementptr inbounds float, float\* %1, i64 %idxprom.i  
%45 = load float, float\* %arrayidx6.i, align 4, !tbaa !12  
%div.i = fdiv float %44, %45, !fpmath !21  
store float %div.i, float\* %arrayidx.i, align 4, !tbaa !12, ... !llvm.access.group !22  
%46 = add nuw i64 %\_local\_id\_x.0, 1  
%exitcond.not = icmp eq i64 %46, %umax  
br i1 %exitcond.not, label %adi\_kernel5.exit.loopexit, label ... %pre\_\nregion\_for\_entry.entry.i, !llvm.loop !27

T

F

adi\_kernel5.exit.loopexit:  
br label %adi\_kernel5.exit

adi\_kernel5.exit:  
ret void

CFG for '\_pocl\_kernel\_adi\_kernel5' function