```
%8:
 %mul.i.i = shl i64 %5, 8
 %sub.i = add i32 %3, -2
 %cmp258.i = icmp slt i32 %3, 3
 %9 = sext i32 %sub.i to i64
 %wide.trip.count.i = zext i32 %sub.i to i64
 %min.iters.check.i = icmp ult i32 %sub.i, 8
 %10 = add nsw i64 %wide.trip.count.i, -1
 %11 = trunc i64 %10 to i32
 %12 = icmp ugt i64 %10, 4294967295
 %mul6.i = tail call { i64, i1 } @llvm.umul.with.overflow.i64(i64 %10, i64 4)
 ... #2
 %mul.result7.i = extractvalue { i64, i1 } %mul6.i, 0
 %mul.overflow8.i = extractvalue { i64, i1 } %mul6.i, 1
 %n.vec.i = and i64 %wide.trip.count.i, 4294967288
 %13 = getelementptr inbounds float, float* %2, i64 -7
 %14 = getelementptr inbounds float, float* %0, i64 -7
 %15 = getelementptr inbounds float, float* %1, i64 -7
 %cmp.n.i = icmp eq i64 %n.vec.i, %wide.trip.count.i
 br label %pregion for entry.entry.i
pregion for entry.entry.i:
\%_local_id_x.0 = phi i64 [ 0, %8 ], [ %83, %if.end.i ] %add1.i.i = add nuw nsw i64 %_local_id_x.0, %mul.i.i, !llvm.access.group !12
%conv.i = trunc i64 %add1.i.i to i32, !llvm.access.group !12
%cmp.i = icmp sge i32 %conv.i, %3, !llvm.access.group !12
%brmerge = or i1 %cmp.i, %cmp258.i
br i1 %brmerge, label %if.end.i, label %for.body.lr.ph.i, !llvm.access.group
...!12
                                     for.body.lr.ph.i:
                                     %mul.i = mul nsw i32 %conv.i, %3, !llvm.access.group !12 %sub21.i = add i32 %mul.i, %3, !llvm.access.group !12
                                     %sub22.i = add i32 %sub21.i, -3, !llvm.access.group !12 %16 = sext i32 %mul.i to i64, !llvm.access.group !12
                                      %sub9.i = add i32 %mul.i, -1, !llvm.access.group !12
                                      br i1 %min.iters.check.i, label %for.body.i.preheader, label
                                     ... %vector.scevcheck.i, !llvm.access.group !12
                                                 vector.scevcheck.i:
                                                 %17 = add i32 %sub.i, %mul.i, !llvm.access.group !12 %18 = sub i32 %17, %11, !llvm.access.group !12 %19 = icmp sgt i32 %18, %17, !llvm.access.group !12 %20 = sub i32 %sub22.i, %11, !llvm.access.group !12
                                                  %21 = icmp sgt i32 %20, %sub22.i, !llvm.access.group !12
                                                  %22 = or i1 %12, %21, !llvm.access.group !12
                                                  %23 = or i1 %22, %19, !llvm.access.group !12
                                                  %24 = sext i32 %17 to i64, !llvm.access.group !12
                                                  %scevgep.i = getelementptr float, float* %2, i64 %24, !llvm.access.group !12
                                                  %scevgep5.i = ptrtoint float* %scevgep.i to i64, !llvm.access.group !12 %25 = icmp ugt i64 %mul.result7.i, %scevgep5.i, !llvm.access.group !12
                                                  %26 = or i1 %mul.overflow8.i, %25, !llvm.access.group !12
                                                  %27 = or i1 %23, %26, !llvm.access.group !12
                                                  %28 = add nsw i64 %16, %9, !llvm.access.group !12
                                                  %scevgep9.i = getelementptr float, float* %2, i64 %28, !llvm.access.group !12
                                                  %scevgep910.i = ptrtoint float* %scevgep9.i to i64, !llvm.access.group !12 %29 = icmp ugt i64 %mul.result7.i, %scevgep910.i, !llvm.access.group !12
                                                  %30 = or i1 %29, %27, !llvm.access.group !12
                                                  %31 = sext i32 %sub22.i to i64, !llvm.access.group !12
                                                  %scevgep14.i = getelementptr float, float* %2, i64 %31, !llvm.access.group
                                                  %scevgep1415.i = ptrtoint float* %scevgep14.i to i64, !llvm.access.group !12
                                                  %32 = icmp ugt i64 %mul.result7.i, %scevgep1415.i, !llvm.access.group !12
                                                  %33 = or i1 %32, %30, !llvm.access.group !12
                                                  %scevgep19.i = getelementptr float, float* %0, i64 %31, !llvm.access.group
                                                  %scevgep1920.i = ptrtoint float* %scevgep19.i to i64, !llvm.access.group !12
                                                  %34 = icmp ugt i64 %mul.result7.i, %scevgep1920.i, !llvm.access.group !12
                                                  %35 = or i1 %mul.overflow8.i, %34, !llvm.access.group !12
                                                  %36 = or i1 %35, %33, !llvm.access.group !12
                                                  %scevgep24.i = getelementptr float, float* %1, i64 %31, !llvm.access.group
                                                  %scevgep2425.i = ptrtoint float* %scevgep24.i to i64, !llvm.access.group !12
                                                  %37 = icmp ugt i64 %mul.result7.i, %scevgep2425.i, !llvm.access.group !12
                                                  %38 = or i1 %37, %36, !llvm.access.group !12
                                                  br i1 %38, label %for.body.i.preheader, label %vector.memcheck.i,
                                                  ...!llvm.access.group!12
                                                                          vector.memcheck.i:
                                                                           %39 = add nsw i64 %24, 1, !llvm.access.group !12
                                                                           %40 = sub nsw i64 %39, %wide.trip.count.i, !llvm.access.group !12
                                                                           %scevgep29.i = getelementptr float, float* %2, i64 %40, !llvm.access.group
                                                                           %scevgep31.i = getelementptr float, float* %2, i64 %39, !llvm.access.group
                                                                           ... !12
                                                                           %41 = add nsw i64 %28, 1, !llvm.access.group !12
                                                                           %42 = sub nsw i64 %41, %wide.trip.count.i, !llvm.access.group !12
                                                                           %scevgep33.i = getelementptr float, float* %2, i64 %42, !llvm.access.group
                                                                           %scevgep35.i = getelementptr float, float* %2, i64 %41, !llvm.access.group
                                                                           ...!12
                                                                           %43 = add nsw i64 %31, 1, !llvm.access.group !12
                                                                           %44 = sub nsw i64 %43, %wide.trip.count.i, !llvm.access.group !12
                                                                           %scevgep37.i = getelementptr float, float* %2, i64 %44, !llvm.access.group
                                                                           ... !12
                                                                           %scevgep39.i = getelementptr float, float* %2, i64 %43, !llvm.access.group
                                                                           %scevgep41.i = getelementptr float, float* %0, i64 %44, !llvm.access.group
                                                                           %scevgep43.i = getelementptr float, float* %0, i64 %43, !llvm.access.group
                                                                           %scevgep45.i = getelementptr float, float* %1, i64 %44, !llvm.access.group
                                                                           %scevgep47.i = getelementptr float, float* %1, i64 %43, !llvm.access.group
                                                                           %bound0.i = icmp ult float* %scevgep29.i, %scevgep35.i, !llvm.access.group
                                                                           %bound1.i = icmp ult float* %scevgep33.i, %scevgep31.i, !llvm.access.group
                                                                           %found.conflict.i = and i1 %bound0.i, %bound1.i, !llvm.access.group !12
                                                                           %bound049.i = icmp ult float* %scevgep29.i, %scevgep39.i, !llvm.access.group
                                                                           %bound150.i = icmp ult float* %scevgep37.i, %scevgep31.i, !llvm.access.group
                                                                           %found.conflict51.i = and i1 %bound150.i, %bound049.i, !llvm.access.group !12
                                                                           %conflict.rdx.i = or i1 %found.conflict.i, %found.conflict51.i,
                                                                           ...!llvm.access.group!12
                                                                           %bound052.i = icmp ult float* %scevgep29.i, %scevgep43.i, !llvm.access.group
                                                                           %bound153.i = icmp ult float* %scevgep41.i, %scevgep31.i, !llvm.access.group
                                                                           %found.conflict54.i = and i1 %bound153.i, %bound052.i, !llvm.access.group !12
                                                                           %conflict.rdx55.i = or i1 %found.conflict54.i, %conflict.rdx.i,
                                                                           ...!llvm.access.group!12
                                                                           %bound056.i = icmp ult float* %scevgep29.i, %scevgep47.i, !llvm.access.group
                                                                           %bound157.i = icmp ult float* %scevgep45.i, %scevgep31.i, !llvm.access.group
                                                                           %found.conflict58.i = and i1 %bound157.i, %bound056.i, !llvm.access.group !12
                                                                           %conflict.rdx59.i = or i1 %found.conflict58.i, %conflict.rdx55.i,
                                                                           ...!llvm.access.group!12
                                                                           br i1 %conflict.rdx59.i, label %for.body.i.preheader, label
                                                                          ... %vector.body.i.preheader, !llvm.access.group !12
                                                                                                                                                        F
                                                                                                                                    vector.body.i.preheader:
                                                                                                                                    br label %vector.body.i
                                                                                           vector.body.i:
                                                                                           %index.next.i1 = phi i64 [ %index.next.i, %vector.body.i ], [ 0,
                                                                                           ... %vector.body.i.preheader ]
                                                                                           %45 = sub nsw i64 %9, %index.next.i1, !llvm.access.group !12
                                                                                           %46 = add nsw i64 %45, %16, !llvm.access.group !12
                                                                                           %47 = getelementptr inbounds float, float* %13, i64 %46, !llvm.access.group
                                                                                           ... !12
                                                                                           %48 = bitcast float* %47 to <8 x float>*, !llvm.access.group !12
                                                                                           %wide.load.i = load <8 x float>, <8 x float>* %48, align 4, !tbaa !14,
                                                                                           ...!alias.scope!18,!llvm.access.group!12
                                                                                           %reverse.i = shufflevector <8 x float> %wide.load.i, <8 x float> undef, <8 x
                                                                                           ... i32> <i32 7, i32 6, i32 5, i32 4, i32 3, i32 2, i32 1, i32 0>,
                                                                                           ...!llvm.access.group!12
                                                                                           %49 = trunc i64 %45 to i32, !llvm.access.group !12
                                                                                           %50 = add i32 %sub9.i, %49, !llvm.access.group !12
                                                                                           %51 = sext i32 %50 to i64, !llvm.access.group !12
                                                                                           %52 = getelementptr inbounds float, float* %13, i64 %51, !llvm.access.group
                                                                                           %53 = bitcast float* %52 to <8 x float>*, !llvm.access.group !12
                                                                                           %wide.load60.i = load <8 x float>, <8 x float>* %53, align \frac{1}{4}, !tbaa !14,
                                                                                           ...!alias.scope!21,!llvm.access.group!12
                                                                                           %reverse61.i = shufflevector <8 x float> %wide.load60.i, <8 x float> undef,
                                                                                           ... < 8 \times i32 > < i32 7, i32 6, i32 5, i32 4, i32 3, i32 2, i32 1, i32 0 > 7
                                                                                           ...!llvm.access.group!12
                                                                                           %54 = trunc i64 %index.next.i1 to i32, !llvm.access.group !12
                                                                                           %55 = sub i32 %3, %54, !llvm.access.group !12
                                                                                           %56 = add i32 %55, %mul.i, !llvm.access.group !12
                                                                                           %57 = add i32 %56, -3, !llvm.access.group !12
                                                                                           %58 = sext i32 %57 to i64, !llvm.access.group !12
                                                                                           %59 = getelementptr inbounds float, float* %14, i64 %58, !llvm.access.group
                                                                                           ... !12
                                                                                           %60 = bitcast float* %59 to <8 x float>*, !llvm.access.group !12
                                                                                           %wide.load62.i = load <8 x float>, <8 x float>* %60, align 4, !tbaa !14,
                                                                                           ...!alias.scope!23,!llvm.access.group!12
                                                                                           %reverse63.i = shufflevector <8 x float> %wide.load62.i, <8 x float> undef,
                                                                                           ... <8 x i32> <i32 7, i32 6, i32 5, i32 4, i32 3, i32 2, i32 1, i32 0>,
                                                                                           ...!llvm.access.group!12
                                                                                           %61 = fneg <8 x float> %reverse61.i, !llvm.access.group !12
                                                                                           \%62 = tail call < 8 x float > @llvm.fmuladd.v8f32(< 8 x float > \%61, < 8 x float >
                                                                                           ... %reverse63.i, <8 x float> %reverse.i) #2, !llvm.access.group !12
                                                                                           %63 = sub i32 %sub22.i, %54, !llvm.access.group !12
                                                                                           %64 = sext i32 %63 to i64, !llvm.access.group !12
                                                                                           %65 = getelementptr inbounds float, float* %15, i64 %64, !llvm.access.group
                                                                                           ... !12
                                                                                           %66 = bitcast float* %65 to <8 x float>*, !llvm.access.group !12
                                                                                           %wide.load64.i = load <8 x float>, <8 x float>* %66, align 4, !tbaa !14,
                                                                                           ...!alias.scope!25,!llvm.access.group!12
                                                                                           %reverse65.i = shufflevector <8 x float> %wide.load64.i, <8 x float> undef,
                                                                                           \dots <8 x i32> <i32 7, i32 6, i32 5, i32 4, i32 3, i32 2, i32 1, i32 0>,
                                                                                           ...!llvm.access.group!12
                                                                                           %67 = fdiv <8 x float> %62, %reverse65.i, !fpmath !27, !llvm.access.group !12
                                                                                           %68 = add i32 %56, -2, !llvm.access.group !12
                                                                                           %69 = sext i32 %68 to i64, !llvm.access.group !12
                                                                                           %reverse66.i = shufflevector < 8 \times 10^{-3} \times
                                                                                           ... <i32 7, i32 6, i32 5, i32 4, i32 3, i32 2, i32 1, i32 0>, !llvm.access.group
                                                                                           %70 = getelementptr inbounds float, float* %13, i64 %69, !llvm.access.group
                                                                                           ... !12
                                                                                           %71 = bitcast float* %70 to <8 x float>*, !llvm.access.group !12
                                                                                           store <8 x float> %reverse66.i, <8 x float>* %71, align 4, !tbaa !14,
                                                                                            .. !alias.scope !28, !noalias !30, !llvm.access.group !12
                                                                                           %index.next.i = add i64 %index.next.i1, 8, !llvm.access.group !12
                                                                                           %72 = icmp eq i64 %index.next.i, %n.vec.i, !llvm.access.group !12
                                                                                           br i1 %72, label %middle.block.i, label %vector.body.i, !llvm.loop !31,
                                                                                           ...!llvm.access.group!12
                                                                                    middle.block.i:
                                                                                     br i1 %cmp.n.i, label %if.end.i, label %for.body.i.preheader,
                                                                                    ...!llvm.access.group!12
                           for.body.i.preheader:
                            %indvars.iv.next.i3.ph = phi i64 [ 0, %for.body.lr.ph.i ], [ 0,
                           ... %vector.scevcheck.i ], [0, %vector.memcheck.i], [%n.vec.i, %middle.block.i
                           br label %for.body.i
                            for.body.i:
                            %indvars.iv.next.i3 = phi i64 [ %indvars.iv.next.i, %for.body.i ], [
                            ... %indvars.iv.next.i3.ph, %for.body.i.preheader ]
                            %73 = sub nsw i64 %9, %indvars.iv.next.i3, !llvm.access.group !12
                            %74 = add nsw i64 %73, %16, !llvm.access.group !12
                            %arrayidx.i = getelementptr inbounds float, float* %2, i64 %74,
                             ..!llvm.access.group!12
                            %75 = load float, float* %arrayidx.i, align 4, !tbaa !14, !llvm.access.group
                            ... !12
                            %76 = trunc i64 %73 to i32, !llvm.access.group !12
                            %add10.i = add i32 %sub9.i, %76, !llvm.access.group !12
                            %idxprom11.i = sext i32 %add10.i to i64, !llvm.access.group !12
                            %arrayidx12.i = getelementptr inbounds float, float* %2, i64 %idxprom11.i,
                            ...!llvm.access.group!12
                            %77 = load float, float* %arrayidx12.i, align 4, !tbaa !14,
                            ...!llvm.access.group!12
                            %78 = trunc i64 %indvars.iv.next.i3 to i32, !llvm.access.group !12
                            %79 = sub i32 %3, %78, !llvm.access.group !12
                            %sub15.i = add i32 %79, %mul.i, !llvm.access.group !12
                            %add16.i = add i32 %sub15.i, -3, !llvm.access.group !12
                            %idxprom17.i = sext i32 %add16.i to i64, !llvm.access.group !12
                            %arravidx18.i = getelementptr inbounds float, float* %0, i64 %idxprom17.i,
                             ..!llvm.access.group!12
                            %80 = load float, float* %arrayidx18.i, align 4, !tbaa !14,
                            ...!llvm.access.group!12
                            %neg.i = fneg float %77, !llvm.access.group !12
                            %81 = tail call float @llvm.fmuladd.f32(float %neg.i, float %80, float %75)
                             .. #2, !llvm.access.group !12
                            %add23.i = sub i32 %sub22.i, %78, !llvm.access.group !12
                            %idxprom24.i = sext i32 %add23.i to i64, !llvm.access.group !12
                            %arravidx25.i = getelementptr inbounds float, float* %1, i64 %idxprom24.i,
                             ..!llvm.access.group!12
                            %82 = load float, float* %arrayidx25.i, align 4, !tbaa !14,
                             ..!llvm.access.group!12
                            %div.i = fdiv float %81, %82, !fpmath !27, !llvm.access.group !12
                            %add29.i = add i32 %sub15.i, -2, !llvm.access.group !12
                            %idxprom30.i = sext i32 %add29.i to i64, !llvm.access.group !12
                            %arravidx31.i = getelementptr inbounds float, float* %2, i64 %idxprom30.i.
                             ..!llvm.access.group!12
                            store float %div.i, float* %arravidx31.i, align 4, !tbaa !14,
                            ...!llvm.access.group!12
                            %indvars.iv.next.i = add nuw nsw i64 %indvars.iv.next.i3, 1,
                             .. !llvm.access.group !12
                            %exitcond.not.i = icmp eq i64 %indvars.iv.next.i, %wide.trip.count.i,
                             ..!llvm.access.group!12
                            br i1 %exitcond.not.i, label %if.end.i.loopexit, label %for.body.i,
                            ...!llvm.loop!34,!llvm.access.group!12
                                                             if.end.i.loopexit:
                                                              br label %if.end.i
                                                                                  if.end.i:
                                                                                   \%83 = add nuw nsw i64 \% local id x.0, 1
                                                                                   \%exitcond.not = icmp eq i\overline{6}4 \%8\overline{3}, \overline{2}56
                                                                                   br i1 %exitcond.not, label %adi kernel3.exit, label
                                                                                  ... %pregion for entry.entry.i, !llvm.loop !35
                                                                                       adi kernel3.exit:
```