```
%10:
                           %11 = \text{sext i} 32 \% 4 \text{ to i} 64
                           %12 = icmp slt i64 %11, 256
                           %13 = select i1 %12, i64 %11, i64 256
                           %mul.i.i = shl i64 %7, 8
                           %mul2.i = mul nsw i32 %5, %3, !llvm.access.group !12
                           %add3.i = add nsw i32 %mul2.i, %3, !llvm.access.group !12
                           %idxprom4.i = sext i32 %add3.i to i64, !llvm.access.group !12
                           %arrayidx5.i = getelementptr inbounds float, float* %1, i64 %idxprom4.i,
                           ...!llvm.access.group!12
                           %14 = icmp ugt i64 \%13, 1
                           %umax = select i1 %14, i64 %13, i64 1
                           %min.iters.check = icmp ult i64 %umax, 8
                           br i1 %min.iters.check, label %pregion for entry.entry.i.preheader, label
                           ... %vector.scevcheck
                                                   Τ
                                                                                                         F
                                                         vector.scevcheck:
                                                          %ident.check = icmp ne i32 %5, 1
                                                          %15 = add nsw i64 \%umax, -1
                                                          %16 = \text{trunc } i64 \%7 \text{ to } i32
                                                          %17 = \text{shl i} 32 \%16, 8
                                                          %18 = add i32 \%17, \%3
                                                          %19 = trunc i64 %15 to i32
                                                          %20 = add i32 %18, %19
                                                          %21 = icmp slt i32 %20, %18
                                                          %22 = icmp ugt i64 %15, 4294967295
                                                          %23 = \text{ or i } 1 \%21, \%22
                                                          %24 = or i1 %ident.check, %23
                                                          br i1 %24, label %pregion for entry.entry.i.preheader, label %vector.ph
                                                                              vector.ph:
                                                                               %n.vec = and i64 %umax, -8
                                                                               %broadcast.splatinsert = insertelement <8 x i64> undef, i64 %mul.i.i, i32 0
                                                                               %broadcast.splat = shufflevector <8 x i64> %broadcast.splatinsert, <8 x i64>
                                                                              ... undef, <8 x i32> zeroinitializer
                                                                               %broadcast.splatinsert1 = insertelement <8 x i32> undef, i32 %5, i32 0
                                                                               %broadcast.splat2 = shufflevector <8 x i32> %broadcast.splatinsert1, <8 x
                                                                              ... i32> undef, <8 x i32> zeroinitializer
                                                                               %broadcast.splatinsert3 = insertelement <8 x i32> undef, i32 %3, i32 0
                                                                               %broadcast.splat4 = shufflevector <8 x i32> %broadcast.splatinsert3, <8 x
                                                                               ... i32> undef, <8 x i32> zeroinitializer
                                                                               %broadcast.splatinsert5 = insertelement <8 x float*> undef, float*
                                                                              ... %arrayidx5.i, i32 0
                                                                               %broadcast.splat6 = shufflevector <8 x float*> %broadcast.splatinsert5, <8 x
                                                                              ... float*> undef, <8 x i32> zeroinitializer
                                                                              br label %vector.body
                                                                           vector.body:
                                                                            %index = phi i64 [ 0, %vector.ph ], [ %index.next, %vector.body ]
                                                                            %vec.ind = phi <8 x i64> [ <i64 0, i64 1, i64 2, i64 3, i64 4, i64 5, i64 6,
                                                                            ... i64 7>, %vector.ph ], [ %vec.ind.next, %vector.body ]
                                                                            %25 = add <8 x i64> %vec.ind, %broadcast.splat, !llvm.access.group !12
                                                                            %26 = trunc <8 x i64> %25 to <8 x i32>, !llvm.access.group !12
                                                                            %27 = mul nsw <8 x i32> %broadcast.splat2, %26, !llvm.access.group !12
                                                                            %28 = add nsw <8 x i32> %27, %broadcast.splat4, !llvm.access.group !12
                                                                            %29 = \text{sext} < 8 \times i32 > %28 \text{ to} < 8 \times i64 >, !llvm.access.group !12
                                                                            %30 = \text{getelementptr inbounds float, float* } \%0, <8 \times i64 > \%29,
                                                                            ...!llvm.access.group!12
                                                                            %wide.masked.gather = call <8 x float> @llvm.masked.gather.v8f32.v8p0f32(<8
                                                                            ... x float*> %30, i32 4, <8 x i1> <i1 true, i1 
                                                                           ... i1 true, i1 true, i1 true, <8 x float> undef), !tbaa !14, !llvm.access.group
                                                                           ...!12
                                                                            %wide.masked.gather7 = call <8 x float> @llvm.masked.gather.v8f32.v8p0f32(<8
                                                                            ... x float*> %broadcast.splat6, i32 4, <8 \times i1> <i1 \text{ true}, i1 true, i1 true, i1
                                                                           ... true, i1 true, i1 true, i1 true, i1 true, <8 x float> undef), !tbaa !14,
                                                                            ...!llvm.access.group!12
                                                                            %31 = fdiv <8 x float> %wide.masked.gather, %wide.masked.gather7, !fpmath
                                                                            ... !18, !llvm.access.group !12
                                                                            %32 = getelementptr inbounds float, float* %2, <8 x i64> %29,
                                                                            ...!llvm.access.group!12
                                                                            call void @llvm.masked.scatter.v8f32.v8p0f32(<8 x float> %31, <8 x float*>
                                                                            ... %32, i32 4, <8 x i1> <i1 true, i1 true, i1 true, i1 true, i1 true, i1 true, i1 true,
                                                                            ... i1 true, i1 true>), !tbaa !14, !llvm.access.group !12
                                                                            %index.next = add i64 %index, 8
                                                                            %vec.ind.next = add <8 x i64> %vec.ind, <i64 8, i64 8, i64 8, i64 8, i64 8,
                                                                            ... i64 8, i64 8, i64 8>
                                                                            %33 = icmp eq i64 %index.next, %n.vec
                                                                            br i1 %33, label %middle.block, label %vector.body, !llvm.loop !19
                                                                                                                                                                    F
                                                                 middle.block:
                                                                  %cmp.n = icmp eq i64 %umax, %n.vec
                                                                  br i1 %cmp.n, label %gramschmidt kernel2.exit, label
                                                                  ... %pregion for entry.entry.i.preheader
  pregion for entry.entry.i.preheader:
   % local id x.0.ph = phi i64 [ 0, %vector.scevcheck ], [ 0, %10 ], [ %n.vec,
   ... %middle.block l
   br label %pregion for entry.entry.i
pregion for entry.entry.i:
\%_{local\_id\_x.0} = phi i64 [ \%36, \%pregion_for_entry.entry.i ], [
... % local id x.0.ph, %pregion for entry.entry.i.preheader
%add1.i.i = add 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
%mul.i = mul nsw i32 %conv.i, %5, !llvm.access.group !12 %add.i = add nsw i32 %mul.i, %3, !llvm.access.group !12
%idxprom.i = sext i32 %add.i to i64, !llvm.access.group !12
%arrayidx.i = getelementptr inbounds float, float* %0, i64 %idxprom.i,
 ..!llvm.access.group!12
%34 = load float, float* %arrayidx.i, align 4, !tbaa !14, !llvm.access.group
%35 = load float, float* %arrayidx5.i, align 4, !tbaa !14,
...!llvm.access.group!12
%div.i = fdiv float %34, %35, !fpmath !18, !llvm.access.group !12
%arrayidx9.i = getelementptr inbounds float, float* %2, i64 %idxprom.i,
...!llvm.access.group!12
store float %div.i, float* %arrayidx9.i, align 4, !tbaa !14,
...!llvm.access.group!12
%36 = add nuw i64\% local id x.0, 1
%exitcond.not = icmp eq i6\overline{4} %\overline{36}, %umax
br i1 %exitcond.not, label %gramschmidt kernel2.exit.loopexit, label
... %pregion for entry.entry.i, !llvm.loop!22
                                                                              F
                           gramschmidt kernel2.exit.loopexit:
                            br label %gramschmidt kernel2.exit
```

gramschmidt kernel2.exit:

... !12