```
%..lcssa.ph.ex phi.pocl context.i = alloca [1 x [1 x [256 x float]]], align
                                                                                                                                                  %11 = bitcast [1 x [1 x [256 x float]]]* %..lcssa.ph.ex phi.pocl context.i
                                                                                                                                                  ... to i8*
                                                                                                                                                  call void @llvm.lifetime.start.p0i8(i64 1024, i8* nonnull %11)
                                                                                                                                                  %12 = shl i64 \%7, 8
                                                                                                                                                  %cmp227.i = icmp sgt i32 %4, 0
%13 = sext i32 %5 to i64
                                                                                                                                                  %14 = \text{sext i} 32 \% 3 \text{ to i} 64
                                                                                                                                                  %wide.trip.count.i = zext i32 %4 to i64
                                                                                                                                                  %15 = add nsw i64 %wide.trip.count.i, -1
                                                                                                                                                  %xtraiter.i = and i64 %wide.trip.count.i, 7
                                                                                                                                                  %16 = icmp ult i64 %15, 7
                                                                                                                                                  %unroll iter.i = sub nuw nsw i64 %wide.trip.count.i, %xtraiter.i
                                                                                                                                                  %lcmp.mod.i = icmp eq i64 %xtraiter.i, 0
                                                                                                                                                  %mul\( \frac{1}{2} \) 0.i = mul ns\( \widetilde{w} \) i3\( \frac{1}{2} \) %5, %3
                                                                                                                                                  %add11.i = add nsw i32 %mul10.i, %3
                                                                                                                                                  %idxprom12.i = sext i32 %add11.i to i64
                                                                                                                                                  %arrayidx13.i = getelementptr inbounds float, float* %1, i64 %idxprom12.i
                                                                                                                                                  br label %pregion for entry.entry.i
                                                                                                                                                         pregion for entry.entry.i:
                                                                                                                                                         % local_id_x.0 = phi i64 [ 0, %10 ], [ %62, %if.end.r_exit.i ] %17 = add nuw nsw i64 %_local_id_x.0, %12
                                                                                                                                                          %conv.i = trunc i64 %17 to i32
                                                                                                                                                          %cmp.i = icmp eq i32 %conv.i, 0
                                                                                                                                                          br i1 %cmp.i, label %for.cond.preheader.i, label %if.end.r exit.i
                                                                                                                            for.cond.preheader.i:
                                                                                                                            br i1 %cmp227.i, label %for.body.preheader.i, label %for.end.i
                                                                                for.body.preheader.i:
                                                                                 br i1 %16, label
                                                                                ... %for.body.preheader.i.for.end.loopexit.unr-lcssa.i crit edge, label
                                                                                ... %for.body.i.preheader
                                                                                                                                             F
                                                                                                                                 for.body.i.preheader:
                                                                                                                                  br label %for.body.i
                                                                                               for.bodv.i:
                                                                                               %niter.nsub.7.i7 = phi i64 [ %niter.nsub.7.i, %for.body.i ], [
                                                                                                 %unroll_iter.i, %for.body.i.preheader ]
                                                                                               \%18 = ph\bar{i} \text{ float } [\%50, \%for.body.i], [0.000000e+00, \%for.body.i.preheader]
                                                                                               %indvars.iv.next.7.i2 = phi i64 [ %indvars.iv.next.7.i, %for.body.i ], [ 0,
                                                                                               ... %for.body.i.preheader ]
                                                                                               %19 = mul nsw i64 %indvars.iv.next.7.i2, %13
                                                                                               %20 = add nsw i64 %19, %14
                                                                                               %arrayidx.i = getelementptr inbounds float, float* %0, i64 %20
                                                                                               %21 = load float, float* %arrayidx.i, align 4, !tbaa !12
                                                                                               %22 = tail call float @llvm.fmuladd.f32(float %21, float %21, float %18) #3
                                                                                               %indvars.iv.next.i = or i64 %indvars.iv.next.7.i2, 1
                                                                                               %23 = mul nsw i64 %indvars.iv.next.i, %13
                                                                                               %24 = add nsw i64 %23, %14
                                                                                                %arrayidx.1.i = getelementptr inbounds float, float* %0, i64 %24
                                                                                                %25 = load float, float* %arrayidx.1.i, align 4, !tbaa !12
                                                                                               %26 = tail call float @llvm.fmuladd.f32(float %25, float %25, float %22) #3
                                                                                               %indvars.iv.next.1.i = or i64 %indvars.iv.next.7.i2, 2
                                                                                                %27 = mul nsw i64 %indvars.iv.next.1.i, %13
                                                                                               %28 = add nsw i64 %27, %14
                                                                                               %arrayidx.2.i = getelementptr inbounds float, float* %0, i64 %28
                                                                                               %29 = load float, float* %arrayidx.2.i, align 4, !tbaa !12
                                                                                               %30 = tail call float @llvm.fmuladd.f32(float %29, float %29, float %26) #3
                                                                                               %indvars.iv.next.2.i = or i64 %indvars.iv.next.7.i2, 3
                                                                                               %31 = mul nsw i64 %indvars.iv.next.2.i, %13
                                                                                               %32 = add nsw i64 %31, %14
                                                                                               %arrayidx.3.i = getelementptr inbounds float, float* %0, i64 %32 %33 = load float, float* %arrayidx.3.i, align 4, !tbaa !12
for.body.preheader.i.for.end.loopexit.unr-lcssa.i crit edge:
%.phi.trans.insert = getelementptr [1 x [1 x (\overline{256} \times \overline{10at})]], [1 x (\overline{1} \times \overline{10})], [1 x (\overline{1} \times \overline{10})]
                                                                                               %34 = tail call float @llvm.fmuladd.f32(float %33, float %33, float %30) #3
... x float]]]* %..lcssa.ph.ex_phi.pocl_context.i, i64 0, i64 0, i64 0, i64
                                                                                               %indvars.iv.next.3.i = or i64 %indvars.iv.next.7.i2, 4
... % local id x.0
                                                                                               %35 = mul nsw i64 %indvars.iv.next.3.i, %13
\%.pre17 = load float, float* \%.phi.trans.insert, align 4
                                                                                               %36 = add nsw i64 %35, %14
 br label %for.end.loopexit.unr-lcssa.i
                                                                                                %arrayidx.4.i = getelementptr inbounds float, float* %0, i64 %36
                                                                                                %37 = load float, float* %arrayidx.4.i, align 4, !tbaa !12
                                                                                                %38 = tail call float @llvm.fmuladd.f32(float %37, float %37, float %34) #3
                                                                                               %indvars.iv.next.4.i = or i64 %indvars.iv.next.7.i2, 5
                                                                                               %39 = mul nsw i64 %indvars.iv.next.4.i, %13
                                                                                               %40 = add nsw i64 %39, %14
                                                                                                %arrayidx.5.i = getelementptr inbounds float, float* %0, i64 %40
                                                                                               %41 = load float, float* %arrayidx.5.i, align 4, !tbaa !12
                                                                                               %42 = tail call float @llvm.fmuladd.f32(float %41, float %41, float %38) #3
                                                                                               %indvars.iv.next.5.i = or i64 %indvars.iv.next.7.i2, 6
                                                                                               %43 = mul nsw i64 %indvars.iv.next.5.i, %13
                                                                                               %44 = add nsw i64 %43, %14
                                                                                               %arrayidx.6.i = getelementptr inbounds float, float* %0, i64 %44
                                                                                               %45 = load float, float* %arrayidx.6.i, align 4, !tbaa !12
                                                                                               %46 = tail call float @llvm.fmuladd.f32(float %45, float %45, float %42) #3
                                                                                               %indvars.iv.next.6.i = or i64 %indvars.iv.next.7.i2, 7
                                                                                               %47 = mul nsw i64 %indvars.iv.next.6.i, %13
                                                                                               %48 = add nsw i64 %47, %14
                                                                                               %arrayidx.7.i = getelementptr inbounds float, float* %0, i64 %48
                                                                                               %49 = load float, float* %arrayidx.7.i, align 4, !tbaa !12
                                                                                               %50 = tail call float @llvm.fmuladd.f32(float %49, float %49, float %46) #3
                                                                                               %indvars.iv.next.7.i = add nuw nsw i64 %indvars.iv.next.7.i2, 8
                                                                                                %niter.nsub.7.i = add i64 %niter.nsub.7.i7, -8
                                                                                               %niter.ncmp.7.i = icmp eq i64 %niter.nsub.7.i, 0
                                                                                               br i1 %niter.ncmp.7.i, label %for.end.loopexit.unr-lcssa.loopexit.i, label
                                                                                                ... %for.body.i
                                                                        for.end.loopexit.unr-lcssa.loopexit.i:
                                                                        %.lcssa = phi float [ %50, %for.body.i ]
%indvars.iv.next.7.i.lcssa = phi i64 [ %indvars.iv.next.7.i, %for.body.i ]
%51 = getelementptr [1 x [1 x [256 x float]]], [1 x [1 x [256 x float]]]*
... %..lcssa.ph.ex_phi.pocl_context.i, i64 0, i64 0, i64 0, i64 %_local_id_x.0
store float %.lcssa, float* %51, align 4, !llvm.access.group !16
                                                                         br label %for.end.loopexit.unr-lcssa.i
                                                                       for.end.loopexit.unr-lcssa.i:
                                                                        \%52 = \text{phi float} [0.0000000e+00]
                                                                       ... %for.body.preheader.i.for.end.loopexit.unr-lcssa.i_crit_edge], [ %.lcssa,
                                                                        ... %for.end.loopexit.unr-lcssa.loopexit.i ]
                                                                       %53 = phi i64 [ 0, ... %for.body.preheader.i.for.end.loopexit.unr-lcssa.i_crit_edge ], [ ... %indvars.iv.next.7.i.lcssa, %for.end.loopexit.unr-lcssa.loopexit.i ]
                                                                        %54 = phi float [ %.pre17,
                                                                       ... %for.body.preheader.i.for.end.loopexit.unr-lcssa.i_crit_edge ], [ %.lcssa, ... %for.end.loopexit.unr-lcssa.loopexit.i ] br i1 %lcmp.mod.i, label %for.end.i, label %for.body.epil.i.preheader
                                                                                                                              for.body.epil.i.preheader:
                                                                                                                              br label %for.body.epil.i
                                                                                                 for.body.epil.i:
                                                                                                 %epil.iter.sub.i13 = phi i64 [ %epil.iter.sub.i, %for.body.epil.i ], [ ... %xtraiter.i, %for.body.epil.i.preheader ]
                                                                                                  %55 = phi float [ %59, %for.body.epil.i ], [ %52, %for.body.epil.i.preheader
                                                                                                 %indvars.iv.next.epil.i9 = phi i64 [ %indvars.iv.next.epil.i, ... %for.body.epil.i ], [ %53, %for.body.epil.i.preheader ] %56 = mul nsw i64 %indvars.iv.next.epil.i9, %13
                                                                                                  %57 = add nsw i64 %56, %14
                                                                                                 %arrayidx.epil.i = getelementptr inbounds float, float* %0, i64 %57 %58 = load float, float* %arrayidx.epil.i, align 4, !tbaa !12 %59 = tail call float @llvm.fmuladd.f32(float %58, float %58, float %55) #3 %indvars.iv.next.epil.i = add nuw nsw i64 %indvars.iv.next.epil.i9, 1 %epil.iter.sub.i = add nsw i64 %epil.iter.sub.i13, -1
                                                                                                  %epil.iter.cmp.i = icmp eq i64 %epil.iter.sub.i, 0
                                                                                                  br i1 %epil.iter.cmp.i, label %for.end.i.loopexit, label %for.body.epil.i,
                                                                                                  ...!llvm.loop!18
                                                                                                                                                                      F
                                                                                                                for.end.i.loopexit:
                                                                                                                 %.lcssa22 = phi float [ %59, %for.body.epil.i ]
                                                                                                                 br label %for.end.i
                                                                                                         for.end.i:
                                                                                                         %60 = phi float [ %54, %for.end.loopexit.unr-lcssa.i ], [ 0.000000e+00, ... %for.cond.preheader.i ], [ %.lcssa22, %for.end.i.loopexit ]
                                                                                                         %61 = tail call float @llvm.sqrt.f32(float %60) #3
                                                                                                         store float %61, float* %arrayidx13.i, align 4, !tbaa !12,
                                                                                                         ...!llvm.access.group!16
                                                                                                         br label %if.end.r exit.i
                                                                                                                                                         if.end.r exit.i:
                                                                                                                                                         \%62 = add nuw nsw i64 \% local id x.0, 1
                                                                                                                                                         %exitcond = icmp eq i64 \%62, 2\overline{5}6
                                                                                                                                                         br i1 %exitcond, label %gramschmidt kernel1.exit, label
                                                                                                                                                         ... %pregion for entry.entry.i, !llvm.loop !20
                                                                                                                                    gramschmidt kernel1.exit:
                                                                                                                                    call void @llvm.lifetime.end.p0i8(i64 1024, i8* nonnull %11)
                                                                                                                                     ret void
```