```
%10:
                                                   %mul.i.i = shl i64 %7, 8
                                                   %cmp227.i = icmp sgt i32 %4, 0
                                                   %11 = \text{sext i} 32 \% 5 \text{ to i} 64
                                                   %12 = \text{sext i} 32 \% 3 \text{ to i} 64
                                                   %wide.trip.count.i = zext i32 %4 to i64
                                                   %mul10.i = mul nsw i32 %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 ], [ %20, %if.end.r exit.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 eq i32 %conv.i, 0, !llvm.access.group !12
                                                  br i1 %cmp.i, label %for.cond.preheader.i, label %if.end.r exit.i,
                                                  ...!llvm.access.group!12
                                   for.cond.preheader.i:
                                   br i1 %cmp227.i, label %for.body.i.preheader, label %for.end.i,
                                   ...!llvm.access.group!12
                                                                                    F
                                    for.body.i.preheader:
                                     br label %for.body.i
for.body.i:
%13 = phi float [ %17, %for.body.i ], [ 0.000000e+00, %for.body.i.preheader ]
%indvars.iv.next.i1 = phi i64 [ %indvars.iv.next.i, %for.body.i ], [ 0,
... %for.body.i.preheader ]
%14 = mul nsw i64 %indvars.iv.next.i1, %11, !llvm.access.group !12
%15 = add nsw i64 %14, %12, !llvm.access.group !12
%arrayidx.i = getelementptr inbounds float, float* %0, i64 %15,
...!llvm.access.group!12
%16 = load float, float* %arrayidx.i, align 4, !tbaa !14, !llvm.access.group
... !12
%17 = tail call float @llvm.fmuladd.f32(float %16, float %16, float %13) #3,
...!llvm.access.group!12
%indvars.iv.next.i = add nuw nsw i64 %indvars.iv.next.i1, 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 %for.end.i.loopexit, label %for.body.i,
...!llvm.loop!18,!llvm.access.group!12
              for.end.i.loopexit:
               %.lcssa = phi float [ %17, %for.body.i ]
               br label %for.end.i
                       for.end.i:
                       %18 = phi float [ 0.000000e+00, %for.cond.preheader.i ], [ %.lcssa,
                       ... %for.end.i.loopexit ]
                       %19 = tail call float @llvm.sqrt.f32(float %18) #3, !llvm.access.group !12
                       store float %19, float* %arrayidx13.i, align 4, !tbaa !14,
                       ...!llvm.access.group!12
                       br label %if.end.r exit.i, !llvm.access.group !12
                                                         if.end.r exit.i:
                                                          %20 = add nuw nsw i64 % local id x.0, 1
                                                          %exitcond.not = icmp eq i\overline{6}4 %20, \overline{2}56
                                                          br i1 %exitcond.not, label %gramschmidt kernel1.exit, label
                                                          ... %pregion for entry.entry.i, !llvm.loop !20
                                                                        Т
                                                                                                         F
                                                          gramschmidt kernel1.exit:
                                                           ret void
```