```
%8:
                                    \%9 = \text{sext i} 32 \%2 \text{ to i} 64
                                    %10 = icmp slt i64 \%9, 32
                                    %11 = select i1 %10, i64 %9, i64 32
                                    %12 = \text{sext i} 32 \% 3 \text{ to i} 64
                                    %13 = icmp slt i64 %12, 8
                                    %14 = select i1 %13, i64 %12, i64 8
                                    %mul.i.i = shl i64 %5, 5
                                    %mul3.i.i = shl i64 %6, 3
                                    %15 = icmp ugt i64 \%11, 1
                                    %umax = select i1 %15, i64 %11, i64 1
                                    %16 = icmp ugt i64 %14, 1
                                    %umax1 = select i1 %16, i64 %14, i64 1
                                    %17 = add nsw i64 %umax, -1
                                    %18 = \text{trunc } i64 \%5 \text{ to } i32
                                    %19 = \text{shl i} 32 \%18, 5
                                    %20 = trunc i64 \%6 to i32
                                    %21 = mul i32 %20, %2
                                    %22 = shl i32 \%21, 3
                                    %23 = \text{zext i} 32 \%22 \text{ to i} 64
                                    %24 = \text{zext i} 32 \% 19 \text{ to i} 64
                                    %25 = add nuw nsw i64 %23, %24
                                    %26 = zext i32 \%2 to i64
                                    br label %pregion for entry.pregion for init.i
                pregion for entry.pregion for init.i:
                 %_local_id_y.0 = phi i64 [0, \%8], [ %75, %pregion_for_end.i ]
                 \%\bar{2}7 = \bar{m}u\bar{l} i64 \% local id y.0, \%26
                 %28 = add i64 %25, %27
                 %29 = trunc i64 %28 to i32
                 %add6.i.i = add i64 % local id_y.0, %mul3.i.i, !llvm.access.group !12
                 %conv2.i = trunc i64 %add6.i.i to i32, !llvm.access.group !12
                 %mul.i = mul nsw i32 %conv2.i, %2, !llvm.access.group !12
                 %min.iters.check = icmp ult i64 %umax, 32
                 br i1 %min.iters.check, label %pregion for entry.entry.i.preheader, label
                 ... %vector.scevcheck
                                        Τ
                                                                                           F
                                           vector.scevcheck:
                                            %30 = \text{trunc } i64 \%17 \text{ to } i32
                                            %31 = add i32 %19, %30
                                            %32 = icmp slt i32 %31, %19
                                            %33 = icmp ugt i64 %17, 4294967295
                                            %34 = \text{ or i } 1 \%32, \%33
                                            %35 = \text{trunc } i64 \%17 \text{ to } i32
                                            %36 = add i32 %29, %35
                                            %37 = icmp slt i32 %36, %29
                                            %38 = icmp ugt i64 %17, 4294967295
                                            %39 = \text{ or i } 1 \%37, \%38
                                            %40 = \text{ or i } 1 \%34, \%39
                                            br i1 %40, label %pregion for entry.entry.i.preheader, label %vector.ph
                                                                                                                     \mathbf{F}
                                                                                               vector.ph:
                                                                                               %n.vec = and i64 %umax, -32
                                                                                               br label %vector.body
                                                             vector.body:
                                                             %index = phi i64 [ 0, %vector.ph ], [ %index.next, %vector.body ]
                                                             %41 = add i64 %index, %mul.i.i, !llvm.access.group !12
                                                             %42 = trunc i64 %41 to i32, !llvm.access.group !12
                                                             %43 = shl i64 %41, 32, !llvm.access.group !12
                                                             %44 = ashr exact i64 %43, 32, !llvm.access.group !12
                                                             %45 = getelementptr inbounds float, float* %0, i64 %44, !llvm.access.group
                                                             ...!12
                                                             %46 = bitcast float* %45 to <8 x float>*
                                                             %wide.load = load <8 x float>, <8 x float>* \%46, align 4, !tbaa !15,
                                                             ...!llvm.access.group!12
                                                             %47 = getelementptr inbounds float, float* %45, i64 8
                                                             %48 = bitcast float* %47 to <8 x float>*
                                                             %wide.load6 = load <8 x float>, <8 x float>* %48, align 4, !tbaa !15,
                                                             ...!llvm.access.group!12
                                                             %49 = getelementptr inbounds float, float* %45, i64 16
                                                             %50 = bitcast float* %49 to <8 x float>*
                                                             \text{wide.load7} = \text{load} < 8 \text{ x float} > \text{, } < 8 \text{ x float} > \text{* } \%50, \text{ align 4, !tbaa !15, }
                                                             ...!llvm.access.group!12
                                                             %51 = getelementptr inbounds float, float* %45, i64 24
                                                             \%52 = bitcast float* \%51 to <8 x float>*
                                                             %wide.load8 = load <8 x float>, <8 x float>* %52, align 4, !tbaa !15,
                                                             ...!llvm.access.group!12
                                                             %53 = add nsw i32 %mul.i, %42, !llvm.access.group !12
                                                             %54 = sext i32 %53 to i64, !llvm.access.group !12
                                                             %55 = getelementptr inbounds float, float* %1, i64 %54, !llvm.access.group
                                                             ... !12
                                                             %56 = bitcast float* %55 to <8 x float>*
                                                             %wide.load9 = load <8 x float>, <8 x float>* %56, align 4, !tbaa !15,
                                                             ...!llvm.access.group!12
                                                             %57 = getelementptr inbounds float, float* %55, i64 8
                                                             \%58 = bitcast float* \%57 to <8 x float>*
                                                             \%wide.load10 = load <8 x float>, <8 x float>* \%58, align 4, !tbaa !15,
                                                             ...!llvm.access.group!12
                                                             %59 = getelementptr inbounds float, float* %55, i64 16
                                                             \%60 = bitcast float* \%59 to <8 x float>*
                                                             %wide.load11 = load <8 x float>, <8 x float>* %60, align 4, !tbaa !15,
                                                             ...!llvm.access.group!12
                                                             %61 = getelementptr inbounds float, float* %55, i64 24
                                                             \%62 = bitcast float* \%61 to <8 x float>*
                                                             %wide.load12 = load <8 x float>, <8 x float>* \%62, align 4, !tbaa !15,
                                                             ...!llvm.access.group!12
                                                             %63 = fsub <8 x float> %wide.load9, %wide.load, !llvm.access.group !12
                                                             %64 = fsub <8 x float> %wide.load10, %wide.load6, !llvm.access.group !12
                                                             %65 = fsub <8 x float> %wide.load11, %wide.load7, !llvm.access.group !12
                                                             %66 = fsub <8 x float> %wide.load12, %wide.load8, !llvm.access.group !12
                                                             \%67 = bitcast float* \%55 to <8 x float>*
                                                             store <8 x float> %63, <8 x float>* %67, align 4, !tbaa !15,
                                                             ...!llvm.access.group!12
                                                             \%68 = bitcast float* \%57 to <8 x float>*
                                                             store <8 x float> %64, <8 x float>* %68, align 4, !tbaa !15,
                                                             ...!llvm.access.group!12
                                                             \%69 = \text{bitcast float*} \%59 \text{ to } < 8 \text{ x float>*}
                                                             store <8 x float> %65, <8 x float>* %69, align 4, !tbaa !15,
                                                             ...!llvm.access.group!12
                                                             \%70 = \text{bitcast float} \%61 \text{ to } < 8 \text{ x float} > *
                                                             store <8 x float> %66, <8 x float>* %70, align 4, !tbaa !15,
                                                             ...!llvm.access.group!12
                                                             %index.next = add i64 %index, 32
                                                             %71 = icmp eq i64 %index.next, %n.vec
                                                             br i1 %71, label %middle.block, label %vector.body, !llvm.loop !19
                                                         middle.block:
                                                          %cmp.n = icmp eq i64 %umax, %n.vec
                                                          br i1 %cmp.n, label %pregion for end.i, label
                                                          ... %pregion for entry.entry.i.preheader
    pregion for entry.entry.i.preheader:
     % local id x.0.ph = phi i64 [0, %vector.scevcheck], 
     ... %pregion for entry pregion for init.i ], [ %n.vec, %middle.block ]
     br label %pregion for entry.entry.i
pregion for entry.entry.i:
\frac{1}{2} \[\text{Nocal_id_x.0 = phi i64 [ \%74, \%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
%sext.i = shl i64 %add1.i.i, 32, !llvm.access.group !12
%idxprom.i = ashr exact i64 %sext.i, 32, !llvm.access.group !12
%arrayidx.i = getelementptr inbounds float, float* %0, i64 %idxprom.i,
..!llvm.access.group!12
%72 = load float, float* %arrayidx.i, align 4, !tbaa !15, !llvm.access.group
...!12
%add.i = add nsw i32 %mul.i, %conv.i, !llvm.access.group !12
%idxprom6.i = sext i32 %add.i to i64, !llvm.access.group !12
%arrayidx7.i = getelementptr inbounds float, float* %1, i64 %idxprom6.i,
..!llvm.access.group!12
%73 = load float, float* %arrayidx7.i, align 4, !tbaa !15,
...!llvm.access.group!12
%sub.i = fsub float %73, %72, !llvm.access.group !12 store float %sub.i, float* %arrayidx7.i, align 4, !tbaa !15,
...!llvm.access.group!12
\%74 = \text{add nuw } i64\% \text{ local id } x.0, 1
%exitcond.not = icmp eq i6\overline{4} %74, %umax
br i1 %exitcond.not, label %pregion for end.i.loopexit, label
... %pregion for entry.entry.i, !llvm.loop 122
                                                                            F
                                    pregion for end.i.loopexit:
                                     br label %pregion for end.i
                                                                           pregion for end.i:
                                                                            \%75 = add \text{ nuw } i64 \% \text{ local } id y.0, 1
                                                                            %exitcond2.not = icmp eq i\overline{64} %75, %umax1
                                                                            br i1 %exitcond2.not, label %reduce kernel.exit, label
                                                                            ... %pregion for entry pregion for init.i, !llvm.loop !23
```