```
%9:
%10 = \text{sext i} 32 \% 4 \text{ to i} 64
%11 = icmp slt i64 %10, 256
%12 = select i1 %11, i64 %10, i64 256
%mul.i.i = shl i64 %6, 8
%mul.i = mul nsw i32 %4, %3
%sub.i = add nsw i32 %3, -1
%mul2.i = mul nsw i32 %sub.i, %4
%13 = icmp ugt i64 \%12, 1
%umax = select i1 %13, i64 %12, 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
                                                         F
                    vector.scevcheck:
                    %14 = add nsw i64 %umax, -1
                    %15 = mul i32 %4, %3
                    %16 = \text{trunc } i64 \%6 \text{ to } i32
                    %17 = \text{shl i} 32 \%16, 8
                    %18 = add nsw i32 %15, %17
                    %19 = \text{trunc } i64 \%14 \text{ to } i32
                    %20 = add i32 %18, %19
                    %21 = icmp slt i32 %20, %18
                    %22 = icmp ugt i64 %14, 4294967295
                    %23 = \text{ or i } 1 \% 21, \%22
                    %24 = add i32 %3, -1
                    %25 = mul i32 %24, %4
                    %26 = add \text{ nsw } i32 \%25, \%17
                    %27 = \text{trunc } i64 \%14 \text{ to } i32
                    %28 = add i32 %26, %27
                    %29 = icmp slt i32 %28, %26
                    %30 = icmp ugt i64 %14, 4294967295
                    %31 = \text{ or i } 1 \% 29, \% 30
                    %32 = \text{ or i } 1 \%23, \%31
                    br i1 %32, label %pregion for entry.entry.i.preheader, label %vector.memcheck
                                              vector.memcheck:
                                              %33 = mul i32 %4, %3
                                              %34 = \text{sext i} 32 \% 33 \text{ to i} 64
                                              %35 = \text{trunc } i64 \%6 \text{ to } i32
                                              %36 = shl i32 %35, 8
                                              %37 = \text{sext i} 32 \% 36 \text{ to i} 64
                                              %38 = add nsw i64 %34, %37
                                              %scevgep = getelementptr float, float* %2, i64 %38
                                              %39 = add nsw i64 %umax, %34
                                              %40 = add nsw i64 %39, %37
                                              %scevgep5 = getelementptr float, float* %2, i64 %40
                                              %scevgep7 = getelementptr float, float* %1, i64 %38
                                               %scevgep9 = getelementptr float, float* %1, i64 %40
                                              %41 = add i32 \%3, -1
                                               %42 = mul i32 %41, %4
                                              %43 = \text{sext i} 32 \% 42 \text{ to i} 64
                                               %44 = add nsw i64 %43, %37
                                               %scevgep11 = getelementptr float, float* %2, i64 %44
                                              %45 = add nsw i64 %umax, %43
                                              %46 = add nsw i64 %45, %37
                                              %scevgep13 = getelementptr float, float* %2, i64 %46 %scevgep15 = getelementptr float, float* %0, i64 %38
                                              %scevgep17 = getelementptr float, float* %0, i64 %40
                                              %scevgep19 = getelementptr float, float* %1, i64 %44
                                              %scevgep21 = getelementptr float, float* %1, i64 %46
                                              %bound0 = icmp ult float* %scevgep, %scevgep9
%bound1 = icmp ult float* %scevgep7, %scevgep5
                                              %found.conflict = and i1 %bound0, %bound1
                                              %bound023 = icmp ult float* %scevgep, %scevgep13
                                              %bound124 = icmp ult float* %scevgep11, %scevgep5
                                              %found.conflict25 = and i1 %bound023, %bound124
                                              %conflict.rdx = or i1 %found.conflict, %found.conflict25
                                              %bound026 = icmp ult float* %scevgep, %scevgep17
                                              %bound127 = icmp ult float* %scevgep15, %scevgep5
                                              %found.conflict28 = and i1 %bound026, %bound127
                                              %conflict.rdx29 = or i1 %conflict.rdx, %found.conflict28
                                              %bound030 = icmp ult float* %scevgep, %scevgep21
                                              %bound131 = icmp ult float* %scevgep19, %scevgep5
                                              %found.conflict32 = and i1 %bound030, %bound131
                                              %conflict.rdx33 = or i1 %conflict.rdx29, %found.conflict32
                                              %bound034 = icmp ult float* %scevgep7, %scevgep13
                                              %bound135 = icmp ult float* %scevgep11, %scevgep9
                                              %found.conflict36 = and i1 %bound034, %bound135
                                              %conflict.rdx37 = or i1 %conflict.rdx33, %found.conflict36
                                              %bound038 = icmp ult float* %scevgep7, %scevgep17
                                              %bound139 = icmp ult float* %scevgep15, %scevgep9
                                              %found.conflict40 = and i1 %bound038, %bound139
                                              %conflict.rdx41 = or i1 %conflict.rdx37, %found.conflict40 %bound042 = icmp ult float* %scevgep7, %scevgep21
                                              %bound143 = icmp ult float* %scevgep19, %scevgep9
                                              %found.conflict44 = and i1 %bound042, %bound143
                                              %conflict.rdx45 = or i1 %conflict.rdx41, %found.conflict44
                                              br i1 %conflict.rdx45, label %pregion for entry.entry.i.preheader, label
                                              ... %vector.ph
                                                               Τ
                                                                                        vector.ph:
                                                                                        %n.vec = and i64 %umax, -8
                                                                                        br label %vector.body
                                                                 vector.body:
                                                                 %index = phi i64 [ 0, %vector.ph ], [ %index.next, %vector.body ]
                                                                 %47 = add i64 %index, %mul.i.i
                                                                 %48 = \text{trunc } i64 \%47 \text{ to } i32
                                                                 %49 = add nsw i32 %mul.i, %48
                                                                 %50 = \text{sext i} 32 \% 49 \text{ to i} 64
                                                                 %51 = getelementptr inbounds float, float* %2, i64 %50
                                                                 \%52 = bitcast float* \%51 to <8 x float>*
                                                                 %wide.load = load < 8 \times float >, < 8 \times float > %52, align 4, !tbaa !12,
                                                                 ... !alias.scope !16, !noalias !19
                                                                 %53 = add nsw i32 %mul2.i, %48
                                                                 %54 = \text{sext i} 32 \% 53 \text{ to i} 64
                                                                 %55 = getelementptr inbounds float, float* %2, i64 %54
                                                                 %56 = bitcast float* %55 to <8 x float>*
                                                                 %wide.load46 = load <8 x float>, <8 x float>* %56, align 4, !tbaa !12,
                                                                 ...!alias.scope!24
                                                                 %57 = getelementptr inbounds float, float* %0, i64 %50
                                                                 %58 = bitcast float* %57 to <8 x float>*
                                                                 %wide.load47 = load <8 x float>, <8 x float>* %58, align 4, !tbaa !12,
                                                                 ... !alias.scope !25
                                                                 %59 = fmul <8 x float> %wide.load46, %wide.load47
                                                                 %60 = getelementptr inbounds float, float* %1, i64 %54
                                                                 %61 = bitcast float* %60 to <8 x float>*
                                                                 %wide.load48 = load <8 x float>, <8 x float>* %61, align 4, !tbaa !12,
                                                                 ... !alias.scope !26
                                                                 %62 = fdiv <8 x float> %59, %wide.load48, !fpmath !27
                                                                 \%63 = \text{fsub} < 8 \times \text{float} > \% \text{wide.load}, \%62
                                                                 \%64 = bitcast float* \%51 to <8 x float>*
                                                                 store <8 x float> %63, <8 x float>* %64, align 4, !tbaa !12, !alias.scope
                                                                 ... !16, !noalias !19, !llvm.access.group !28
                                                                 %65 = getelementptr inbounds float, float* %1, i64 %50
                                                                 \%66 = bitcast float* \%65 to <8 x float>*
                                                                 %wide.load49 = load <8 x float>, <8 x float>* \%66, align 4, !tbaa !12,
                                                                 ... !alias.scope !30, !noalias !31
                                                                 \%67 = \text{bitcast float*} \%57 \text{ to } < 8 \text{ x float>*}
                                                                 %wide.load50 = load <8 x float>, <8 x float>* %67, align 4, !tbaa !12,
                                                                 ... !alias.scope !25
                                                                 \%68 = \text{fmul} < 8 \times \text{float} > \% \text{wide.load50}, \% \text{wide.load50}
                                                                 \%69 = \text{bitcast float*} \%60 \text{ to } < 8 \text{ x float>*}
                                                                 %wide.load51 = load <8 x float>, <8 x float>* %69, align 4, !tbaa !12,
                                                                 ... !alias.scope !26
                                                                 %70 = fdiv <8 x float> %68, %wide.load51, !fpmath !27
                                                                 %71 = fsub <8 x float> %wide.load49, %70
                                                                 \%72 = bitcast float* \%65 to <8 x float>*
                                                                 store <8 x float> %71, <8 x float>* %72, align 4, !tbaa !12, !alias.scope
                                                                 ... !30, !noalias !31, !llvm.access.group !28
                                                                 %index.next = add i64 %index, 8
                                                                 %73 = icmp eq i64 %index.next, %n.vec
                                                                 br i1 %73, label %middle.block, label %vector.body, !llvm.loop !32
                                                            middle.block:
                                                            %cmp.n = icmp eq i64 %umax, %n.vec
                                                            br i1 %cmp.n, label %adi kernel4.exit, label
                                                            ... %pregion for entry.entry.i.preheader
pregion for entry.entry.i.preheader:
% local id x.0.ph = phi i64 [ 0, %vector.memcheck ], [ 0, %vector.scevcheck
... ], [0, \( \sqrt{9} \) ], [%n.vec, %middle.block ]
br label %pregion for entry.entry.i
pregion for entry.entry.i:
\%[local[id]x.0] = phi i64 [ \%81, \%pregion for entry.entry.i ], [
... % local_id_x.0.ph, %pregion for entry.entry.i.preheader ]
%add1.i.i = add i64 % local id x.\overline{0}, %mul.i.i
%conv.i = trunc i64 %add1.i.i to i32
%add.i = add nsw i32 %mul.i, %conv.i
%idxprom.i = sext i32 %add.i to i64
%arrayidx.i = getelementptr inbounds float, float* %2, i64 %idxprom.i %74 = load float, float* %arrayidx.i, align 4, !tbaa !12
%add3.i = add nsw i32 %mul2.i, %conv.i
%idxprom4.i = sext i32 %add3.i to i64
%arrayidx5.i = getelementptr inbounds float, float* %2, i64 %idxprom4.i
%75 = load float, float* %arrayidx5.i, align 4, !tbaa !12
%arrayidx9.i = getelementptr inbounds float, float* %0, i64 %idxprom.i
%76 = load float, float* %arrayidx9.i, align 4, !tbaa !12
%mul10.i = fmul float %75, %76
%arrayidx15.i = getelementptr inbounds float, float* %1, i64 %idxprom4.i
%77 = load float, float* %arrayidx15.i, align 4, !tbaa !12
%div.i = fdiv float %mul10.i, %77, !fpmath !27
%sub16.i = fsub float %74, %div.i
store float %sub16.i, float* %arrayidx.i, align 4, !tbaa !12,
...!llvm.access.group!28
%arrayidx24.i = getelementptr inbounds float, float* %1, i64 %idxprom.i
%78 = load float, float* %arrayidx24.i, align 4, !tbaa !12 %79 = load float, float* %arrayidx9.i, align 4, !tbaa !12
%mul33.i = fmul float %79, %79
%80 = load float, float* %arrayidx15.i, align 4, !tbaa !12
%div39.i = fdiv float %mul33.i, %80, !fpmath !27
%sub40.i = fsub float %78, %div39.i
store float %sub40.i, float* %arrayidx24.i, align 4, !tbaa !12,
...!llvm.access.group!28
\%81 = \text{add nuw } i64\% \text{ local id } x.0, 1
%exitcond.not = icmp eq i6\overline{4} %81, %umax
br i1 %exitcond.not, label %adi kernel4.exit.loopexit, label
... %pregion for entry.entry.i, !llvm.loop !35
                                                           F
                             adi kernel4.exit.loopexit:
                             br label %adi kernel4.exit
```