```
%11 = \text{sext i} 32 \% 5 \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
                                                                                                               %cmp228.i = icmp sgt i32 %5, 0, !llvm.access.group !12
                                                                                                               %14 = zext i32 %5 to i64
                                                                                                               %15 = icmp ugt i64 %13, 1
                                                                                                               %umax = select i1 %15, 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.ph
                                                                                                                                              Τ
                                                                                                                                                                                                                      F
                                                                                                                                                                                   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.splatinsert11 = insertelement <8 x i64> undef, i64 %14, i32 0
                                                                                                                                                                                    %broadcast.splat12 = shufflevector <8 x i64> %broadcast.splatinsert11, <8 x
                                                                                                                                                                                    ... i64> undef. <8 x i32> zeroinitializer
                                                                                                                                                                                     %broadcast.splatinsert14 = insertelement <8 x float> undef, float %4, i32 0
                                                                                                                                                                                    %broadcast.splat15 = shufflevector <8 x float> %broadcast.splatinsert14, <8
                                                                                                                                                                                    ... x float> undef. <8 x i32> zeroinitializer
                                                                                                                                                                                    br label %vector.body
                                                                                                                                                                                                       vector.body:
                                                                                                                                                                                                       %index = phi i64 [ 0, %vector.ph ], [ %index.next, %for.end.i19 ]
                                                                                                                                                                                                       % vec.ind = phi < 8 \times i64 > [ < i64 0, i64 1, i64 2, i64 3, i64 4, i64 5, i64 6, i64 6]
                                                                                                                                                                                                       ... i64 7>, %vector.ph ], [ %vec.ind.next, %for.end.i19 ]
                                                                                                                                                                                                        %16 = add <8 x i64> %vec.ind, %broadcast.splat, !llvm.access.group !12
                                                                                                                                                                                                       %17 = shl < 8 \times i64 > %16, < i64 32, i64
                                                                                                                                                                                                        ... i64 32, i64 32>, !llvm.access.group !12
                                                                                                                                                                                                       \%18 = ashr exact < 8 \times i64 > \%17, < i64 32, i64 32, i64 32, i64 32, i64 32, i64 32, i64
                                                                                                                                                                                                       ... 32, i64 32, i64 32>, !llvm.access.group !12
                                                                                                                                                                                                        %19 = getelementptr inbounds float, float* %1, <8 x i64> %18,
                                                                                                                                                                                                       ...!llvm.access.group!12
                                                                                                                                                                                                       %wide.masked.gather8 = call <8 x float> @llvm.masked.gather.v8f32.v8p0f32(<8
                                                                                                                                                                                                       ... x float*> %19, i32 4, <8 x i1> <i1 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
                                                                                                                                                                                                       br i1 %cmp228.i, label %for.body.i9.preheader, label %for.end.i19
                                                                                                                                                                                                                                           Τ
                                                                                                                                                                                                                  for.body.i9.preheader:
                                                                                                                                                                                                                   br label %for.body.i9
                                                                                                                                                  for.body.i9:
                                                                                                                                                   \text{%vec.phi} = \text{phi} < 8 \times i64 > [\%27, \% \text{for.body.i9}], [zeroinitializer,]
                                                                                                                                                   ... %for.body.i9.preheader ]
                                                                                                                                                   %vec.phi10 = phi < 8 x float > [ %26, %for.body.i9 ], [ %wide.masked.gather8,
                                                                                                                                                   ... %for.body.i9.preheader ]
                                                                                                                                                   %20 = mul nuw nsw <8 x i64> %vec.phi, %broadcast.splat12, !llvm.access.group
                                                                                                                                                   ... !12
                                                                                                                                                   %21 = add nsw <8 x i64> %20, %18, !llvm.access.group !12
                                                                                                                                                   \%22 = \text{getelementptr inbounds float, float* } \%0, <8 \text{ x } i64 > \%21,
                                                                                                                                                   ...!llvm.access.group!12
                                                                                                                                                   %wide.masked.gather13 = call <8 x float>
                                                                                                                                                  ... @llvm.masked.gather.v8f32.v8p0f32(<8 x float*> %22, i32 4, <8 x i1> <i1 true, i1 true, i1
                                                                                                                                                   ... undef), !tbaa !14, !llvm.access.group !12
                                                                                                                                                   %23 = fmul <8 x float> %wide.masked.gather13, %broadcast.splat15,
                                                                                                                                                   ...!llvm.access.group!12
                                                                                                                                                   %24 = getelementptr inbounds float, float* %2, <8 x i64> %vec.phi,
                                                                                                                                                   ...!llvm.access.group!12
                                                                                                                                                   %wide.masked.gather16 = call <8 x float>
                                                                                                                                                   ... @llvm.masked.gather.v8f32.v8p0f32(<8 x float*> %24, i32 4, <8 x i1> <i1 true,
                                                                                                                                                  ... i1 true, i7 true, i1 true,
                                                                                                                                                  ... undef), !tbaa !14, !llvm.access.group !12
                                                                                                                                                  %25 = fmul <8 x float> %23, %wide.masked.gather16, !llvm.access.group !12 %26 = fadd <8 x float> %vec.phi10, %25, !llvm.access.group !12 call void @llvm.masked.scatter.v8f32.v8p0f32(<8 x float> %26, <8 x float*>
                                                                                                                                                  ... %19, 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
                                                                                                                                                   %27 = add nuw nsw <8 x i64> %vec.phi, <i64 1, i64 1, i64 1, i64 1, i64 1,
                                                                                                                                                  ... i64 1, i64 1>, !llvm.access.group !12
%28 = icmp eq <8 x i64> %27, %broadcast.splat12, !llvm.access.group !12
                                                                                                                                                   %29 = extractelement <8 x i1> %28, i32 0
                                                                                                                                                   br i1 %29, label %for.end.i19.loopexit, label %for.body.i9
                                                                                                                                                                                                                                                                         F
                                                                                                                                                                                     for.end.i19.loopexit:
                                                                                                                                                                                      \%.lcssa25 = phi < 8 x float > [\%26, \%for.body.i9]
                                                                                                                                                                                      br label %for.end.i19
                                                                                                                                                                                                       for.end.i19:
                                                                                                                                                                                                        %vec.phi20 = phi <8 x float> [ %wide.masked.gather8, %vector.body ], [
                                                                                                                                                                                                        ... %.lcssa25, %for.end.i19.loopexit ]
                                                                                                                                                                                                        %30 = getelementptr inbounds float, float* %3, <8 x i64> %18,
                                                                                                                                                                                                        ...!llvm.access.group!12
                                                                                                                                                                                                       %wide.masked.gather21 = call <8 x float> ... @llvm.masked.gather.v8f32.v8p0f32(<8 x float*> %30, i32 4, \leq8 x i1> \leqi1 true,
                                                                                                                                                                                                       ... i1 true, <8 x float>
                                                                                                                                                                                                       ... undef), !tbaa !14, !llvm.access.group !12
                                                                                                                                                                                                        %31 = fadd <8 x float> %vec.phi20, %wide.masked.gather21, !llvm.access.group
                                                                                                                                                                                                       ...!12
                                                                                                                                                                                                        call void @llvm.masked.scatter.v8f32.v8p0f32(<8 x float> %31, <8 x float*>
                                                                                                                                                                                                       ... %19, 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>
                                                                                                                                                                                                        %32 = icmp eq i64 %index.next, %n.vec
                                                                                                                                                                                                        br i1 %32, label %middle.block, label %vector.body, !llvm.loop !18
                                                                                                                                                                                               middle.block:
                                                                                                                                                                                                %cmp.n = icmp eq i64 %umax, %n.vec
                                                                                                                                                                                                br i1 %cmp.n, label %gemver kernel2.exit, label
                                                                                                                                                                                                ... %pregion for entry.entry.i.preheader
                                                                           pregion for entry.entry.i.preheader:
                                                                            % local id x.0.ph = phi i64 [ 0, %10 ], [ %n.vec, %middle.block ]
                                                                            br label %pregion for entry.entry.i
                                                                 pregion for entry.entry.i:
                                                                 \%_local_id_x.0 = phi i64 [ %42, %for.end.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 %sext27.i = shl i64 %add1.i.i, 32, !llvm.access.group !12
                                                                 %idxprom8.i = ashr exact i64 %sext27.i, 32, !llvm.access.group !12
                                                                 %arrayidx9.i = getelementptr inbounds float, float* %1, i64 %idxprom8.i,
                                                                   .!llvm.access.group!12
                                                                 %.pre.i = load float, float* %arrayidx9.i, align 4, !tbaa !14,
                                                                    !llvm.access.group !12
                                                                 br i1 %cmp228.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:
%indvars.iv.next.i3 = phi i64 [ %indvars.iv.next.i, %for.body.i ], [ 0,
 .. %for.body.i.preheader ]
%33 = phi float [ %39, %for.body.i ], [ %.pre.i, %for.body.i.preheader ] %34 = mul nuw nsw i64 %indvars.iv.next.i3, %14, !llvm.access.group !12
%35 = add nsw i64 %34, %idxprom8.i, !llvm.access.group !12
%arrayidx.i = getelementptr inbounds float, float* %0, i64 %35,
  ..!llvm.access.group!12
%36 = load float, float* %arrayidx.i, align 4, !tbaa !14, !llvm.access.group
 .. !12
%mul4.i = fmul float %36, %4, !llvm.access.group !12
%arrayidx6.i = getelementptr inbounds float, float* %2, i64
   %indvars.iv.next.i3, !llvm.access.group !12
%37 = load float, float* %arrayidx6.i, align 4, !tbaa !14,
  ..!llvm.access.group!12
%38 = fmul float %mul4.i, %37, !llvm.access.group !12
%39 = fadd float %33, %38, !llvm.access.group !12
store float %39, float* %arrayidx9.i, align 4, !tbaa !14, !llvm.access.group
 ...!12
%indvars.iv.next.i = add nuw nsw i64 %indvars.iv.next.i3, 1,
 ..!llvm.access.group!12
%exitcond.not.i = icmp eq i64 %indvars.iv.next.i, %14, !llvm.access.group !12 br i1 %exitcond.not.i, label %for.end.i.loopexit, label %for.body.i,
 ...!llvm.loop!21,!llvm.access.group!12
                                                                                                                F
                                       for.end.i.loopexit:
                                       %.lcssa = phi float [ %39, %for.body.i ]
                                       br label %for.end.i
                                                         for.end.i:
                                                          %40 = phi float [ %.pre.i, %pregion for entry.entry.i ], [ %.lcssa,
                                                          ... %for.end.i.loopexit ]
                                                          %arrayidx11.i = getelementptr inbounds float, float* %3, i64 %idxprom8.i,
                                                          ... !llvm.access.group !12
%41 = load float, float* %arrayidx11.i, align 4, !tbaa !14,
                                                          ...!llvm.access.group!12
                                                          %add14.i = fadd float %40, %41, !llvm.access.group !12 store float %add14.i, float* %arrayidx9.i, align 4, !tbaa !14,
                                                         "...!llvm.access.group!12

%42 = add nuw i64 %_local_id_x.0, 1
%exitcond.not = icmp eq i64 %42, %umax
br i1 %exitcond.not, label %gemver_kernel2.exit.loopexit, label
... %pregion_for_entry.entry.i,!llvm.loop!23
                                                                                                        gemver kernel2.exit.loopexit:
                                                                                                         br label %gemver_kernel2.exit
                                                                                                                                     gemver kernel2.exit:
```

%10: