```
%..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
                                                                                                                                             %cmp228.i = icmp sgt i32 %5, 0
                                                                                                                                             %13 = zext i32 %5 to i64
                                                                                                                                             %14 = add nsw i64 %13, -1
                                                                                                                                             %xtraiter.i = and i64 %13, 3
                                                                                                                                             %15 = icmp ult i64 %14, 3
                                                                                                                                             %unroll iter.i = sub nuw nsw i64 %13, %xtraiter.i
                                                                                                                                             %lcmp.mod.i = icmp eq i64 %xtraiter.i, 0
                                                                                                                                             br label %pregion for entry.entry.i
                                                                                                                                                 pregion for entry.entry.i:
                                                                                                                                                  \%\overline{1}6 = \overline{a}d\overline{d} nuw nsw i64 % local id x.0, %12
                                                                                                                                                  %conv.i = trunc i64 %16 to i32
                                                                                                                                                  %cmp.i = icmp slt i32 %conv.i, %5
                                                                                                                                                  br i1 %cmp.i, label %for.cond.preheader.i, label %if.end.r exit.i
                                                                                                       for.cond.preheader.i:
                                                                                                        %sext27.i = shl i64 %16, 32
                                                                                                        %idxprom8.i = ashr exact i64 %sext27.i, 32
                                                                                                        %arrayidx9.i = getelementptr inbounds float, float* %1, i64 %idxprom8.i
                                                                                                        %.pre.i221 = load float, float* %arrayidx9.i, align 4, !tbaa !12
                                                                                                        br i1 %cmp228.i, label %for.body.lr.ph.i, label %for.end.i
                                                                         for.body.lr.ph.i:
                                                                         br i1 %15, label %for.body.lr.ph.i.for.end.loopexit.unr-lcssa.i crit edge,
                                                                         ... label %for.body.i.preheader
                                                                                                                           for.body.i.preheader:
                                                                                                                            br label %for.body.i
                                                                                           for.bodv.i:
                                                                                           %niter.nsub.3.i9 = phi i64 [ %niter.nsub.3.i, %for.body.i ], [
                                                                                            ... %unroll iter.i, %for.body.i.preheader ]
                                                                                            %indvars.iv.next.3.i6 = phi i64 [ %indvars.iv.next.3.i, %for.body.i ], [ 0,
                                                                                           ... %for.body.i.preheader ]
                                                                                            %17 = phi float [ %37, %for.body.i ], [ %.pre.i221, %for.body.i.preheader ]
                                                                                            %18 = mul nuw nsw i64 %indvars.iv.next.3.i6, %13
                                                                                            %19 = add nsw i64 %18, %idxprom8.i
                                                                                           %arrayidx.i = getelementptr inbounds float, float* %0, i64 %19 %20 = load float, float* %arrayidx.i, align 4, !tbaa !12
                                                                                           %mul4.i = fmul float %20, %4
                                                                                           %arrayidx6.i = getelementptr inbounds float, float* %2, i64
                                                                                           ... %indvars.iv.next.3.i6
                                                                                            %21 = load float, float* %arrayidx6.i, align 4, !tbaa !12
                                                                                           %22 = tail call float @llvm.fmuladd.f32(float %mul4.i, float %21, float %17)
                                                                                           store float %22, float* %arrayidx9.i, align 4, !tbaa !12, !llvm.access.group
                                                                                            %indvars.iv.next.i = or i64 %indvars.iv.next.3.i6, 1
                                                                                            %23 = mul nuw nsw i64 %indvars.iv.next.i, %13
                                                                                            %24 = add nsw i64 %23, %idxprom8.i
                                                                                           %arrayidx.1.i = getelementptr inbounds float, float* %0, i64 %24 %25 = load float, float* %arrayidx.1.i, align 4, !tbaa !12
                                                                                            %mul4.1.i = fmul float %25, %4
                                                                                           %arrayidx6.1.i = getelementptr inbounds float, float* %2, i64
                                                                                            ... %indvars.iv.next.i
                                                                                            %26 = load float, float* %arrayidx6.1.i, align 4, !tbaa !12
%27 = tail call float @llvm.fmuladd.f32(float %mul4.1.i, float %26, float
                                                                                           ... %22) #3
for.body.lr.ph.i.for.end.loopexit.unr-lcssa.i_crit_edge:
                                                                                           store float %27, float* %arrayidx9.i, align 4, !tbaa !12, !llvm.access.group
%.phi.trans.insert = 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
                                                                                           ... !16
                                                                                           %indvars.iv.next.1.i = or i64 %indvars.iv.next.3.i6, 2
... % local id x.0
                                                                                            %28 = mul nuw nsw i64 %indvars.iv.next.1.i, %13
%.pre19 = load float, float* %.phi.trans.insert, align 4
                                                                                            %29 = add nsw i64 %28, %idxprom8.i
br label %for.end.loopexit.unr-lcssa.i
                                                                                            %arrayidx.2.i = getelementptr inbounds float, float* %0, i64 %29
                                                                                            %30 = load float, float* %arrayidx.2.i, align 4, !tbaa !12
                                                                                            %mul4.2.i = fmul float %30, %4
                                                                                            %arrayidx6.2.i = getelementptr inbounds float, float* %2, i64
                                                                                           ... %indvars.iv.next.1.i
                                                                                            %31 = load float, float* %arrayidx6.2.i, align 4, !tbaa !12
                                                                                            %32 = tail call float @llvm.fmuladd.f32(float %mul4.2.i, float %31, float
                                                                                            ... %27) #3
                                                                                           store float %32, float* %arrayidx9.i, align 4, !tbaa !12, !llvm.access.group
                                                                                           ... !16
                                                                                            %indvars.iv.next.2.i = or i64 %indvars.iv.next.3.i6, 3
                                                                                            %33 = mul nuw nsw i64 %indvars.iv.next.2.i, %13
                                                                                            %34 = add nsw i64 %33, %idxprom8.i
                                                                                           %arrayidx.3.i = getelementptr inbounds float, float* %0, i64 %34 %35 = load float, float* %arrayidx.3.i, align 4, !tbaa !12
                                                                                            %mul4.3.i = fmul float %35, %4
                                                                                           %arrayidx6.3.i = getelementptr inbounds float, float* %2, i64
                                                                                           ... %indvars.iv.next.2.i
                                                                                            %36 = load float, float* %arrayidx6.3.i, align 4, !tbaa !12
                                                                                           %37 = tail call float @llvm.fmuladd.f32(float %mul4.3.i, float %36, float
                                                                                           ... %32) #3
                                                                                            store float %37, float* %arrayidx9.i, align 4, !tbaa !12, !llvm.access.group
                                                                                           ... !16
                                                                                            %indvars.iv.next.3.i = add nuw nsw i64 %indvars.iv.next.3.i6, 4
                                                                                            %niter.nsub.3.i = add i64 %niter.nsub.3.i9, -4
                                                                                           %niter.ncmp.3.i = icmp eq i64 %niter.nsub.3.i, 0
br i1 %niter.ncmp.3.i, label %for.end.loopexit.unr-lcssa.loopexit.i, label
                                                                                           ... %for.body.i
                                                                     for.end.loopexit.unr-lcssa.loopexit.i:
                                                                     %.lcssa = phi float [ %37, %for.body.i ]
                                                                     %indvars.iv.next.3.i.lcssa = phi i64 [ %indvars.iv.next.3.i, %for.body.i ] %38 = 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 %_local_id_x.0 store float %.lcssa, float* %38, align 4, !llvm.access.group !16
                                                                     br label %for.end.loopexit.unr-lcssa.i
                                                                    for.end.loopexit.unr-lcssa.i:
                                                                     %39 = phi i64 [ 0, %for.body.lr.ph.i.for.end.loopexit.unr-lcssa.i_crit_edge ... ], [ %indvars.iv.next.3.i.lcssa, %for.end.loopexit.unr-lcssa.loopexit.i ]
                                                                     \%40 = phi float [ %.pre.i221,
                                                                    ... %for.body.lr.ph.i.for.end.loopexit.unr-lcssa.i_crit_edge ], [ %.lcssa, ... %for.end.loopexit.unr-lcssa.loopexit.i ] %41 = phi float [ %.pre19,
                                                                    ... %for.body.lr.ph.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.i15 = phi i64 [ %epil.iter.sub.i, %for.body.epil.i ], [ ... %xtraiter.i, %for.body.epil.i.preheader ]
                                                                                             %indvars.iv.next.epil.i13 = phi i64 [ %indvars.iv.next.epil.i,
                                                                                             ... %for.body.epil.i ], [ %39, %for.body.epil.i.preheader ]
                                                                                             %42 = phi float [ %47, %for.body.epil.i ], [ %40, %for.body.epil.i.preheader
                                                                                             %43 = mul nuw nsw i64 %indvars.iv.next.epil.i13, %13
                                                                                             %44 = add nsw i64 %43, %idxprom8.i
                                                                                             %arrayidx.epil.i = getelementptr inbounds float, float* %0, i64 %44 %45 = load float, float* %arrayidx.epil.i, align 4, !tbaa !12
                                                                                             %mul4.epil.i = fmul float %45, %4
                                                                                             %arrayidx6.epil.i = getelementptr inbounds float, float* %2, i64
                                                                                             ... %indvars.iv.next.epil.i13
                                                                                             %46 = load float, float* %arrayidx6.epil.i, align 4, !tbaa !12
%47 = tail call float @llvm.fmuladd.f32(float %mul4.epil.i, float %46, float
                                                                                             ... %42) #3
                                                                                             store float %47, float* %arrayidx9.i, align 4, !tbaa !12, !llvm.access.group
                                                                                             %indvars.iv.next.epil.i = add nuw nsw i64 %indvars.iv.next.epil.i13, 1
                                                                                             %epil.iter.sub.i = add nsw i64 %epil.iter.sub.i15, -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
                                                                                                         for.end.i.loopexit:
                                                                                                          %.lcssa25 = phi float [ %47, %for.body.epil.i ]
                                                                                                          br label %for.end.i
                                                                                               for.end.i:
                                                                                               %48 = phi float [ %41, %for.end.loopexit.unr-lcssa.i ], [ %.pre.i221,
                                                                                               ... %for.cond.preheader.i ], [ %.lcssa25, %for.end.i.loopexit ]
                                                                                               %arrayidx11.i = getelementptr inbounds float, float* %3, i64 %idxprom8.i
                                                                                                %49 = load float, float* %arrayidx11.i, align 4, !tbaa !12
                                                                                                %add14.i = fadd float %48, %49
                                                                                               store float %add14.i, float* %arrayidx9.i, align 4, !tbaa !12,
                                                                                                ...!llvm.access.group!16
                                                                                               br label %if.end.r exit.i
                                                                                                                                                      if.end.r exit.i:
                                                                                                                                                      \%50 = add nuw nsw i64 \% local id x.0, 1
                                                                                                                                                      %exitcond = icmp eq i64 %50, 256
br i1 %exitcond, label %gemver_kernel2.exit, label
                                                                                                                                                      ... %pregion for entry.entry.i, !llvm.loop !20
                                                                                                                                gemver kernel2.exit:
                                                                                                                                 call void @llvm.lifetime.end.p0i8(i64 1024, i8* nonnull %11)
```

ret void