```
%9:
                                                                 %mul.i.i = shl i64 %6, 8
                                                                  %sub.i = sub i32 -2, %3
                                                                 %sub2.i = add i32 %sub.i, %4
                                                                  %mul.i = mul nsw i32 %sub2.i, %4
                                                                  %sub3.i = sub i32 -3, %3
                                                                  %sub4.i = add i32 %sub3.i, %4
                                                                  %mul5.i = mul nsw i32 %sub4.i, %4
                                                                  br label %pregion for entry.entry.i
                                                     pregion for entry.entry.i:
                                                      % local id x.0 = phi i64 [0, %9], [%21, %if.end.r exit.i.1]
                                                      %add1.i.i = add nuw nsw i64 % local id x.0, %mul.i.i
                                                      %conv.i = trunc i64 %add1.i.i to i32
                                                      %cmp.i = icmp slt i32 %conv.i. %4
                                                      br i1 %cmp.i, label %if.then.i, label %if.end.r exit.i
             if.then.i:
             %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
             %10 = load float, float* %arrayidx.i, align 4, !tbaa !12
             %add6.i = add nsw i32 %mul5.i, %conv.i
             %idxprom7.i = sext i32 %add6.i to i64
             %arravidx8.i = getelementptr inbounds float, float* %2, i64 %idxprom7.i
             %11 = load float, float* %arrayidx8.i, align 4, !tbaa !12
             %arrayidx14.i = getelementptr inbounds float, float* %0, i64 %idxprom7.i
             %12 = load float, float* %arrayidx14.i, align 4, !tbaa !12
             %neg.i = fneg float %11
             %13 = tail call float @llvm.fmuladd.f32(float %neg.i, float %12, float %10)
             ... #3
             %arrayidx21.i = getelementptr inbounds float, float* %1, i64 %idxprom.i
             %14 = load float, float* %arrayidx21.i, align 4, !tbaa !12
             %div.i = fdiv float %13, %14, !fpmath !16
             store float %div.i, float* %arrayidx.i, align 4, !tbaa !12,
             ...!llvm.access.group!17
             br label %if.end.r exit.i
                          if.end.r exit.i:
                           %15 = \text{ or } i64 \% \text{ local id } x.0, 1
                           %add1.i.i.1 = add nuw nsw i64 %15, %mul.i.i
                           %conv.i.1 = trunc i64 %add1.i.i.1 to i32
                           %cmp.i.1 = icmp slt i32 %conv.i.1, %4
                           br i1 %cmp.i.1, label %if.then.i.1, label %if.end.r exit.i.1
                                       Т
                                                                     \mathbf{F}
%add.i.1 = add nsw i32 %mul.i, %conv.i.1
%idxprom.i.1 = sext i32 %add.i.1 to i64
%arrayidx.i.1 = getelementptr inbounds float, float* %2, i64 %idxprom.i.1
%16 = load float, float* %arrayidx.i.1, align 4, !tbaa !12
%add6.i.1 = add nsw i32 %mul5.i, %conv.i.1
%idxprom7.i.1 = sext i32 %add6.i.1 to i64
%arravidx8.i.1 = getelementptr inbounds float, float* %2, i64 %idxprom7.i.1
%17 = load float, float* %arrayidx8.i.1, align 4, !tbaa !12
%arrayidx14.i.1 = getelementptr inbounds float, float* %0, i64 %idxprom7.i.1
%18 = load float, float* %arrayidx14.i.1, align 4, !tbaa !12
%neg.i.1 = fneg float %17
%19 = tail call float @llvm.fmuladd.f32(float %neg.i.1, float %18, float
... %16) #3
%arrayidx21.i.1 = getelementptr inbounds float, float* %1, i64 %idxprom.i.1
%20 = load float, float* %arravidx21.i.1, align 4, !tbaa !12
%div.i.1 = fdiv float %19, %20, !fpmath !16
store float %div.i.1, float* %arravidx.i.1, align 4, !tbaa !12,
...!llvm.access.group!17
br label %if.end.r exit.i.1
                                                     if.end.r exit.i.1:
                                                      %21 = add nuw nsw i64 % local id x.0, 2
                                                      %exitcond.not.1 = icmp eq i64 %21, 256
                                                      br i1 %exitcond.not.1, label %adi kernel6.exit, label
                                                      ... %pregion for entry.entry.i, !llvm.loop !19
                                                         adi kernel6.exit:
                                                          ret void
```

if.then.i.1: