```
%mul.i.i = shl i64 %5, 5
                                                                            %mul3.i.i = shl i64 %6, 3
                                                                           %sub.i = add nsw i32 %2, -1, !llvm.access.group !12
                                                                            %sub4.i = add nsw i32 %3, -1
                                                                            br label %pregion for entry.pregion for init.i
                                                              pregion for entry.pregion for init.i:
                                                               % local id y.0 = phi i64 [0, \overline{8}], [\%34, \%pregion for end.i]
                                                               %add6.i.i = add nuw nsw 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
                                                               %cmp.i = icmp sgt i32 %sub.i, %conv2.i, !llvm.access.group !12
                                                               %cmp8.i = icmp sqt i32 %conv2.i, 0
                                                               %sub13.i = add nsw i32 %conv2.i, -1
                                                               %mul.i = mul nsw i32 %sub13.i, %3
                                                               %mul31.i = mul nsw i32 %conv2.i, %3
                                                               %add51.i = add nuw nsw i32 %conv2.i, 1
                                                               %mul52.i = mul nsw i32 %add51.i, %3
                                                               br label %pregion for_entry.i
                                         pregion for entry.entry.i:
                                         % [ocal] id [x.0] = phi i64 [ 0, %pregion for entry.pregion for init.i ], [
                                         ... %35, %if.end.i ]
                                         %add1.i.i = add nuw nsw 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
                                          br i1 %cmp.i, label %land.lhs.true.i, label %if.end.i, !llvm.access.group !12
                   land.lhs.true.i:
                   %cmp5.i = icmp sgt i32 %sub4.i, %conv.i, !llvm.access.group !12
                   %or.cond.i = and i1 %cmp8.i, %cmp5.i, !llvm.access.group !12
                   %cmp11.i = icmp sgt i32 %conv.i, 0, !llvm.access.group !12
                   %or.cond76.i = and i1 %cmp11.i, %or.cond.i, !llvm.access.group !12
                   br i1 %or.cond76.i, label %if.then.i, label %if.end.i, !llvm.access.group !12
if.then.i:
%sub14.i = add nsw i32 %conv.i, -1, !llvm.access.group !12
%add.i = add nsw i32 %sub14.i, %mul.i, !llvm.access.group !12
%idxprom.i = sext i32 %add.i to i64, !llvm.access.group !12
%arrayidx.i = getelementptr inbounds float, float* %0, i64 %idxprom.i,
...!llvm.access.group!12
%9 = load float, float* %arrayidx.i, align 4, !tbaa !15, !llvm.access.group
%add19.i = add nsw i32 %mul.i, %conv.i, !llvm.access.group !12
%idxprom20.i = sext i32 %add19.i to i64, !llvm.access.group !12
%arrayidx21.i = getelementptr inbounds float, float* %0, i64 %idxprom20.i,
...!llvm.access.group!12
%10 = load float, float* %arrayidx21.i, align 4, !tbaa !15,
...!llvm.access.group!12
%mul22.i = fmul float %10, 5.000000e-01, !llvm.access.group !12
%11 = fmul float %9, 0x3FC99999A0000000, !llvm.access.group !12
%12 = fadd float %11, %mul22.i, !llvm.access.group !12
%add25.i = add nuw nsw i32 %conv.i, 1, !llvm.access.group !12
%add26.i = add nsw i32 %add25.i, %mul.i, !llvm.access.group !12
%idxprom27.i = sext i32 %add26.i to i64, !llvm.access.group !12
%arrayidx28.i = getelementptr inbounds float, float* %0, i64 %idxprom27.i,
...!llvm.access.group!12
%13 = load float, float* %arrayidx28.i, align 4, !tbaa !15,
...!llvm.access.group!12
%14 = fmul float %13, 0x3FE99999A0000000, !llvm.access.group !12
%15 = fsub float %12, %14, !llvm.access.group !12
%add33.i = add nsw i32 %sub14.i, %mul31.i, !llvm.access.group !12
%idxprom34.i = sext i32 %add33.i to i64, !llvm.access.group !12
%arrayidx35.i = getelementptr inbounds float, float* %0, i64 %idxprom34.i,
...!llvm.access.group!12
%16 = load float, float* %arrayidx35.i, align 4, !tbaa !15,
...!llvm.access.group!12
%17 = fmul float %16, 0x3FD3333340000000, !llvm.access.group !12
%18 = fsub float %15, %17, !llvm.access.group !12
%add40.i = add nsw i32 %mul31.i, %conv.i, !llvm.access.group !12
%idxprom41.i = sext i32 %add40.i to i64, !llvm.access.group !12
%arrayidx42.i = getelementptr inbounds float, float* %0, i64 %idxprom41.i,
...!llvm.access.group!12
%19 = load float, float* %arrayidx42.i, align 4, !tbaa !15,
...!llvm.access.group!12
%20 = fmul float %19, 0x3FE3333340000000, !llvm.access.group !12
%21 = fadd float %20, %18, !llvm.access.group !12
%add47.i = add nsw i32 %add25.i, %mul31.i, !llvm.access.group !12
%idxprom48.i = sext i32 %add47.i to i64, !llvm.access.group !12
%arrayidx49.i = getelementptr inbounds float, float* %0, i64 %idxprom48.i,
...!llvm.access.group!12
%22 = load float, float* %arrayidx49.i, align 4, !tbaa !15,
...!llvm.access.group!12
%23 = fmul float %22, 0x3FECCCCC0000000, !llvm.access.group !12
%24 = fsub float %21, %23, !llvm.access.group !12
%add54.i = add nsw i32 %sub14.i, %mul52.i, !llvm.access.group !12
%idxprom55.i = sext i32 %add54.i to i64, !llvm.access.group !12
%arrayidx56.i = getelementptr inbounds float, float* %0, i64 %idxprom55.i,
...!llvm.access.group!12
%25 = load float, float* %arrayidx56.i, align 4, !tbaa !15,
...!llvm.access.group!12
%26 = fmul float %25, 0x3FD99999A0000000, !llvm.access.group !12
%27 = fadd float %26, %24, !llvm.access.group !12
%add61.i = add nsw i32 %mul52.i, %conv.i, !llvm.access.group !12
%idxprom62.i = sext i32 %add61.i to i64, !llvm.access.group !12
%arrayidx63.i = getelementptr inbounds float, float* %0, i64 %idxprom62.i,
...!llvm.access.group!12
%28 = load float, float* %arrayidx63.i, align 4, !tbaa !15,
...!llvm.access.group!12
%29 = fmul float %28, 0x3FE6666660000000, !llvm.access.group !12
%30 = fadd float %29, %27, !llvm.access.group !12
%add68.i = add nsw i32 %add25.i, %mul52.i, !llvm.access.group !12
%idxprom69.i = sext i32 %add68.i to i64, !llvm.access.group !12
%arrayidx70.i = getelementptr inbounds float, float* %0, i64 %idxprom69.i,
...!llvm.access.group!12
%31 = load float, float* %arrayidx70.i, align 4, !tbaa !15,
... !llvm.access.group !12
%32 = fmul float %31, 0x3FB99999A0000000, !llvm.access.group !12
%33 = fadd float %32, %30, !llvm.access.group !12
%arrayidx75.i = getelementptr inbounds float, float* %1, i64 %idxprom41.i,
...!llvm.access.group!12
store float %33, float* %arrayidx75.i, align 4, !tbaa !15,
...!llvm.access.group!12
br label %if.end.i, !llvm.access.group !12
                                                           if.end.i:
                                                           %35 = add nuw nsw i64 \% local id x.0, 1
                                                           %exitcond.not = icmp eq i\overline{6}4 %3\overline{5}, \overline{3}2
                                                           br i1 %exitcond.not, label %pregion for end.i, label
                                                           ... %pregion for entry.entry.i, !llvm.loop 122
                                                          pregion for end.i:
                                                          ^{1}\%34 = add nuw nsw i64 % local id v.0, 1
                                                           %exitcond1.not = icmp eq \overline{1}64 %\overline{3}4, 8
                                                           br i1 %exitcond1.not, label %Convolution2D kernel.exit, label
                                                           ... %pregion for entry pregion for init.i, !llvm.loop!19
                                                           Convolution2D kernel.exit:
```

%8:

ret void