```
%8:
                                                                            %mul.i.i = shl i64 %5, 5
                                                                            %mul3.i.i = shl i64 %6, 3
                                                                            %sub.i = add nsw i32 %2, -1
                                                                            %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, \sqrt{8}8], [%27, %pregion for end.i]
                                                                  \%\overline{a}dd6.\overline{i}.i = add nuw nsw i64 \% local id y.0, \%mul\overline{3}.i.i
                                                                  %conv2.i = trunc i64 %add6.i.i to i32
                                                                  %cmp.i = icmp sgt i32 %sub.i, %conv2.i
                                                                  %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 i1 %cmp.i, label %pregion for entry.entry.i.us.preheader, label
                                                                  ... %pregion for end.i
                                                                                                                   F
                                                           pregion for entry.entry.i.us.preheader:
                                                           br label %pregion for entry.entry.i.us
                                             pregion for entry.entry.i.us:
                                             % local id x.0.us = phi i64 [ %26, %if.end.i.us ], [ 0,
                                             ... \(\bar{v}\) pregion_for_entry.entry.i.us.preheader ]
                                             %add1.i.i.us = add nuw nsw i64 % local id x.0.us, %mul.i.i
                                             %conv.i.us = trunc i64 %add1.i.i.us to i32
                                             %cmp5.i.us = icmp sgt i32 %sub4.i, %conv.i.us
                                             %or.cond.i.us = and i1 %cmp8.i, %cmp5.i.us
                                              %cmp11.i.us = icmp sgt i32 %conv.i.us, 0
                                             %or.cond76.i.us = and i1 %cmp11.i.us, %or.cond.i.us
                                             br i1 %or.cond76.i.us, label %if.then.i.us, label %if.end.i.us
if.then.i.us:
%sub14.i.us = add nsw i32 %conv.i.us, -1
%add.i.us = add nsw i32 %sub14.i.us, %mul.i
%idxprom.i.us = sext i32 %add.i.us to i64
%arrayidx.i.us = getelementptr inbounds float, float* %0, i64 %idxprom.i.us
%9 = load float, float* %arrayidx.i.us, align 4, !tbaa !12
%add19.i.us = add nsw i32 %mul.i, %conv.i.us
%idxprom20.i.us = sext i32 %add19.i.us to i64
%arrayidx21.i.us = getelementptr inbounds float, float* %0, i64
... %idxprom20.i.us
%10 = load float, float* %arrayidx21.i.us, align 4, !tbaa !12
%mul22.i.us = fmul float %10, 5.000000e-01
%11 = tail call float @llvm.fmuladd.f32(float %9, float 0x3FC99999A0000000,
... float %mul22.i.us) #3
%add25.i.us = add nuw nsw i32 %conv.i.us, 1
%add26.i.us = add nsw i32 %add25.i.us, %mul.i
%idxprom27.i.us = sext i32 %add26.i.us to i64
%arrayidx28.i.us = getelementptr inbounds float, float* %0, i64
... %idxprom27.i.us
%12 = load float, float* %arrayidx28.i.us, align 4, !tbaa !12
%13 = tail call float @llvm.fmuladd.f32(float %12, float 0xBFE99999A0000000,
... float %11) #3
%add33.i.us = add nsw i32 %sub14.i.us, %mul31.i
%idxprom34.i.us = sext i32 %add33.i.us to i64
%arrayidx35.i.us = getelementptr inbounds float, float* %0, i64
... %idxprom34.i.us
%14 = load float, float* %arrayidx35.i.us, align 4, !tbaa !12
%15 = tail call float @llvm.fmuladd.f32(float %14, float 0xBFD3333340000000,
... float %13) #3
%add40.i.us = add nsw i32 %mul31.i, %conv.i.us
%idxprom41.i.us = sext i32 %add40.i.us to i64
%arrayidx42.i.us = getelementptr inbounds float, float* %0, i64
... %idxprom41.i.us
%16 = load float, float* %arrayidx42.i.us, align 4, !tbaa !12
%17 = tail call float @llvm.fmuladd.f32(float %16, float 0x3FE3333340000000,
... float %15) #3
%add47.i.us = add nsw i32 %add25.i.us, %mul31.i
%idxprom48.i.us = sext i32 %add47.i.us to i64
%arrayidx49.i.us = getelementptr inbounds float, float* %0, i64
... %idxprom48.i.us
%18 = load float, float* %arrayidx49.i.us, align 4, !tbaa !12
%19 = tail call float @llvm.fmuladd.f32(float %18, float 0xBFECCCCC00000000,
... float %17) #3
%add54.i.us = add nsw i32 %sub14.i.us, %mul52.i
%idxprom55.i.us = sext i32 %add54.i.us to i64
%arrayidx56.i.us = getelementptr inbounds float, float* %0, i64
... %idxprom55.i.us
%20 = load float, float* %arrayidx56.i.us, align 4, !tbaa !12
%21 = tail call float @llvm.fmuladd.f32(float %20, float 0x3FD99999A0000000,
... float %19) #3
%add61.i.us = add nsw i32 %mul52.i, %conv.i.us
%idxprom62.i.us = sext i32 %add61.i.us to i64
%arrayidx63.i.us = getelementptr inbounds float, float* %0, i64
... %idxprom62.i.us
%22 = load float, float* %arrayidx63.i.us, align 4, !tbaa !12
%23 = tail call float @llvm.fmuladd.f32(float %22, float 0x3FE6666660000000,
... float %21) #3
%add68.i.us = add nsw i32 %add25.i.us, %mul52.i
%idxprom69.i.us = sext i32 %add68.i.us to i64
%arrayidx70.i.us = getelementptr inbounds float, float* %0, i64
... %idxprom69.i.us
%24 = load float, float* %arrayidx70.i.us, align 4, !tbaa !12
%25 = tail call float @llvm.fmuladd.f32(float %24, float 0x3FB99999A0000000,
... float %23) #3
%arrayidx75.i.us = getelementptr inbounds float, float* %1, i64
... %idxprom41.i.us
store float %25, float* %arrayidx75.i.us, align 4, !tbaa !12,
...!llvm.access.group!16
br label %if.end.i.us
                                          if.end.i.us:
                                          %26 = add nuw nsw i64 % local id x.0.us, 1
                                          \%exitcond.not = icmp eq i64 \%26, 32
                                          br i1 %exitcond.not, label %pregion for end.i.loopexit, label
                                          ... %pregion for entry.entry.i.us, !llvm.loop !19
                                                            pregion for end.i.loopexit:
                                                            br label %pregion for end.i
                                                               pregion for end.i:
                                                               \%27 = add nuw nsw i64 % local id y.0, 1
                                                               \%exitcond2.not = icmp eq \overline{i}64 \% \overline{27}, 8
                                                               br i1 %exitcond2.not, label %Convolution2D kernel.exit, label
                                                               ... %pregion for entry.pregion for init.i, !llvm.loop !21
                                                                Convolution2D kernel.exit:
                                                                ret void
```