```
\%9 = \text{shl } i64 \%4, 5
                                                                                                                                                                                                    %sub.i = add nsw i32 %2, -1
                                                                                                                                                                                                    %10 = trunc i64 %5 to i32
                                                                                                                                                                                                    %11 = mul i32 %10, %2
                                                                                                                                                                                                    %12 = shl i32 %11, 3
                                                                                                                                                                                                    %13 = trunc i64 %4 to i32
                                                                                                                                                                                                    %14 = shl i32 %13, 5
                                                                                                                                                                                                    %15 = add i32 %12, %14
                                                                                                                                                                                                    %16 = \text{zext i} 32 \% 2 \text{ to i} 64
                                                                                                                                                                                                    %17 = add i32 %15, -8
                                                                                                                                                                                                    %18 = \text{ or } i32 \%17, 7
                                                                                                                                                                                                    %19 = \text{ or } i32 \%15, 1
                                                                                                                                                                                                    %20 = \text{shl i} 32 \%10.3
                                                                                                                                                                                                    %21 = \text{ or } i32 \%20, 1
                                                                                                                                                                                                    %22 = mul i32 %21, %2
                                                                                                                                                                                                    %23 = add i32 %22, %14
                                                                                                                                                                                                    %24 = add i32 %20, -1
                                                                                                                                                                                                    %25 = mul i32 %24, %2
                                                                                                                                                                                                    %26 = add i32 %25, %14
                                                                                                                                                                                                    %27 = trunc i64 %5 to i32
                                                                                                                                                                                                    %28 = mul i32 %27, %2
                                                                                                                                                                                                    %29 = \text{shl i} 32 \%28, 3
                                                                                                                                                                                                    %30 = trunc i64 %4 to i32
                                                                                                                                                                                                    %31 = shl i32 %30, 5
                                                                                                                                                                                                    %32 = add i32 %29, %31
                                                                                                                                                                                                    %33 = zext i32 \%2 to i64
                                                                                                                                                                                                    %scevgep17 = getelementptr float, float* %1, i64 32
                                                                                                                                                                                                    %34 = \sinh i32 \%27, 3
                                                                                                                                                                                                    %35 = add i32 %34, -1
                                                                                                                                                                                                    %36 = mul i32 %35, %2
                                                                                                                                                                                                    %37 = add i32 %36, %31
                                                                                                                                                                                                    %scevgep22 = getelementptr float, float* %0, i64 32
                                                                                                                                                                                                    %38 = \text{ or } i32 \%34, 1
                                                                                                                                                                                                    %39 = mul i32 %38, %2
                                                                                                                                                                                                    %40 = add i32 %39, %31
                                                                                                                                                                                                    %scevgep27 = getelementptr float, float* %0, i64 32
                                                                                                                                                                                                    %41 = \text{ or } i32 \%32, 1
                                                                                                                                                                                                    %42 = \text{zext i} 32 \% 41 \text{ to i} 64
                                                                                                                                                                                                    %scevgep32 = getelementptr float, float* %0, i64 32
                                                                                                                                                                                                    %43 = add i32 %29, %31
                                                                                                                                                                                                    %44 = add i32 %43, -8
                                                                                                                                                                                                    %45 = \text{ or i } 32 \%44, 7
                                                                                                                                                                                                    %scevgep37 = getelementptr float, float* %0, i64 32
                                                                                                                                                                                                    %scevgep42 = getelementptr float, float* %0, i64 32
                                                                                                                                                                                                    br label %pregion for entry.pregion for init.i
                                                                                                                                                                                            pregion_for_entry.pregion_for_init.i:
                                                                                                                                                                                            \%_local_id_y.0 = phi i64 [0, \%7], [ \%123, \%pregion_for_end.i ]
                                                                                                                                                                                            \%\overline{4}6 = \overline{m}u\overline{l} i64 \% local id y.0, \%33
                                                                                                                                                                                            %47 = \text{trunc } i64 \ \%46 \ \text{to} \ i32
                                                                                                                                                                                            %48 = add i32 %32, %47
                                                                                                                                                                                            %49 = \text{sext i} 32 \% 48 \text{ to i} 64
                                                                                                                                                                                             %scevgep = getelementptr float, float* %1, i64 %49
                                                                                                                                                                                             %scevgep18 = getelementptr float, float* %scevgep17, i64 %49
                                                                                                                                                                                            %50 = trunc i64 %46 to i32
                                                                                                                                                                                            %51 = add i32 %37, %50
                                                                                                                                                                                            %52 = \text{sext i} 32 \% 51 \text{ to i} 64
                                                                                                                                                                                            %scevgep20 = getelementptr float, float* %0, i64 %52
                                                                                                                                                                                            %scevgep23 = getelementptr float, float* %scevgep22, i64 %52 %53 = trunc i64 %46 to i32
                                                                                                                                                                                            %54 = add i32 %40, %53
                                                                                                                                                                                            %55 = \text{sext i} 32 \% 54 \text{ to i} 64
                                                                                                                                                                                            %scevgep25 = getelementptr float, float* %0, i64 %55
                                                                                                                                                                                             %scevgep28 = getelementptr float, float* %scevgep27, i64 %55
                                                                                                                                                                                            %56 = add i64 %46, %42
                                                                                                                                                                                            %sext = shl i64 %56, 32
                                                                                                                                                                                            %57 = ashr exact i64 %sext, 32
                                                                                                                                                                                             %scevgep30 = getelementptr float, float* %0, i64 %57
                                                                                                                                                                                            %scevgep33 = getelementptr float, float* %scevgep32, i64 %57 %58 = trunc i64 %46 to i32
                                                                                                                                                                                            %59 = add i32 %45, %58
                                                                                                                                                                                            \%60 = \text{sext i} 32 \% 59 \text{ to i} 64
                                                                                                                                                                                            %scevgep35 = getelementptr float, float* %0, i64 %60
%scevgep38 = getelementptr float, float* %scevgep37, i64 %60
%61 = mul i64 %_local_id_y.0, %16
                                                                                                                                                                                            \%62 = add nuw nsw i6\overline{4} \% local_id_y.0, \%8
                                                                                                                                                                                            %conv.i = trunc i64 %62 to i32
                                                                                                                                                                                            %cmp.i = icmp sgt i32 %conv.i, 0
%mul.i = mul nsw i32 %conv.i, %2
                                                                                                                                                                                            %add25.i = add nuw nsw i32 %conv.i, 1
                                                                                                                                                                                            %mul26.i = mul nsw i32 %add25.i, %2
                                                                                                                                                                                            %sub31.i = add nsw i32 %conv.i, -1
                                                                                                                                                                                            %mul32.i = mul nsw i32 %sub31.i, %2
                                                                                                                                                                                            %cmp4.i = icmp sgt i32 %sub.i, %conv.i
                                                                                                                                                                                            %or.cond = and i1 %cmp.i, %cmp4.i
                                                                                                                                                                                            br i1 %or.cond, label %vector.scevcheck, label %pregion for end.i
                                                                                         vector.scevcheck:
                                                                                          %63 = trunc i64 %61 to i32
                                                                                          %64 = add i32 %26, %63
                                                                                          %65 = trunc i64 %61 to i32
                                                                                          %66 = add i32 %23, %65
%67 = trunc i64 %61 to i32
                                                                                          %68 = add i32 %19, %67
                                                                                          %69 = trunc i64 %61 to i32
                                                                                          %70 = add i32 %18, %69
                                                                                          %71 = trunc i64 %61 to i32
                                                                                          %72 = add i32 %15, %71
                                                                                          %73 = icmp sgt i32 %72, 2147483616
                                                                                          %74 = icmp sgt i32 %70, 2147483616
%75 = or i1 %73, %74
                                                                                          %76 = icmp sgt i32 %68, 2147483616
%77 = or i1 %75, %76
                                                                                          %78 = icmp sgt i32 %66, 2147483616
%79 = or i1 %77, %78
                                                                                          %80 = icmp sgt i32 %64, 2147483616
%81 = or i1 %79, %80
                                                                                          br i1 %81, label %pregion_for_entry.entry.i.us.us.preheader, label
                                                                                          ... %vector.memcheck
                                                                                                           vector.memcheck:
                                                                                                           %bound0 = icmp ult float* %scevgep, %scevgep23
%bound1 = icmp ult float* %scevgep20, %scevgep18
%found.conflict = and i1 %bound0, %bound1
                                                                                                           %bound045 = icmp ult float* %scevgep, %scevgep28
%bound146 = icmp ult float* %scevgep25, %scevgep18
%found.conflict47 = and i1 %bound045, %bound146
                                                                                                           %conflict.rdx = or i1 %found.conflict, %found.conflict47
                                                                                                           %bound048 = icmp ult float* %scevgep, %scevgep33
%bound149 = icmp ult float* %scevgep30, %scevgep18
%found.conflict50 = and i1 %bound048, %bound149
                                                                                                           %conflict.rdx51 = or i1 %conflict.rdx, %found.conflict50
                                                                                                           %bound052 = icmp ult float* %scevgep, %scevgep38
%bound153 = icmp ult float* %scevgep35, %scevgep18
%found.conflict54 = and i1 %bound052, %bound153
                                                                                                           %conflict.rdx55 = or i1 %conflict.rdx51, %found.conflict54
                                                                                                           %bound056 = icmp ugt float* %scevgep42, %1
%bound157 = icmp ugt float* %scevgep17, %0
%found.conflict58 = and i1 %bound056, %bound157
%conflict.rdx59 = or i1 %conflict.rdx55, %found.conflict58
                                                                                                           br i1 %conflict.rdx59, label %pregion for entry.entry.i.us.us.preheader,
                                                                                                           ... label %vector.ph
                                                                                                                                 vector.ph:
                                                                                                                                 %broadcast.splatinsert = insertelement <8 x i64> undef, i64 %9, i32 0
                                                                                                                                 %broadcast.splat = shufflevector <8 x i64> %broadcast.splatinsert, <8 x i64>
                                                                                                                                 ... undef, <8 x i32> zeroinitializer
                                                                      pregion for entry.entry.i.us.us.preheader:
                                                                       br label %pregion for entry.entry.i.us.us
                                                                                                                                 %broadcast.splatinsert60 = insertelement <8 x i32> undef, i32 %sub.i, i32 0 %broadcast.splat61 = shufflevector <8 x i32> %broadcast.splatinsert60, <8 x
                                                                                                                                 ... i32> undef, <8 x i32> zeroinitializer
                                                                                                                                 br label %vector.body
                                                                                                                           vector.body:
                                                                                                                           %index = phi i64 [ 0, %vector.ph ], [ %index.next, %vector.body ] %vec.ind = phi <8 x i64> [ <i64 0, i64 1, i64 2, i64 3, i64 4, i64 5, i64 6,
                                                                                                                           ... i64 7>, %vector.ph ], [ %vec.ind.next, %vector.body ] %82 = add nuw nsw <8 x i64> %vec.ind, %broadcast.splat
                                                                                                                           %83 = trunc <8 x i64> %82 to <8 x i32>
                                                                                                                           %84 = icmp sgt <8 x i32> %83, zeroinitializer

%85 = icmp sgt <8 x i32> %broadcast.splat61, %83

%86 = and <8 x i1> %85, %84
                                                                                                                           \%87 = \text{extractelement} < 8 \text{ x i} 32 > \%83, i 32 0
                                                                                                                           %88 = add i32 %mul.i, %87
                                                                                                                           %89 = \text{sext i} 32 \% 88 \text{ to i} 64
                                                                                                                           %90 = getelementptr inbounds float, float* %0, i64 %89 %91 = bitcast float* %90 to <8 x float>*
                                                                                                                           %wide.masked.load = call <8 x float> @llvm.masked.load.v8f32.p0v8f32(<8 x
                                                                                                                           ... float>* %91, i32 4, <8 x i1> %86, <8 x float> undef), !tbaa !12, !alias.scope
                                                                                                                          ... !16
                                                                                                                           \%92 = add i32 \%88, -1
                                                                                                                           \%93 = \text{sext i} 32 \%92 \text{ to i} 64
                                                                                                                           %94 = getelementptr inbounds float, float* %0, i64 %93
                                                                                                                           %95 = bitcast float* %94 to <8 x float>*
                                                                                                                           %wide.masked.load62 = call <8 x float> @llvm.masked.load.v8f32.p0v8f32(<8 x ^{\circ}
                                                                                                                           ... float>* %95, i32 4, <8 x i1> %86, <8 x float> undef), !tbaa !12, !alias.scope
                                                                                                                           %96 = fadd <8 x float> %wide.masked.load, %wide.masked.load62
                                                                                                                           \%97 = add i32 \%88, 1
                                                                                                                           \%98 = \text{sext i} 32 \%97 \text{ to i} 64
                                                                                                                           %99 = getelementptr inbounds float, float* %0, i64 %98
                                pregion for entry.entry.i.us.us:
                                                                                                                           %100 = bitcast float* %99 to <8 x float>*
                                 \%_{local\_id\_x.0.us.us} = phi i64 [\%122,\%if.end.i.us.us], [0,]
                                                                                                                           %wide.masked.load63 = call <8 x float> @llvm.masked.load.v8f32.p0v8f32(<8 x ^{\circ}
                                ... %pregion_for_entry.entry.i.us.us.preheader ]
%116 = add nuw nsw i64 %_local_id_x.0.us.us, %9
                                                                                                                           ... float>* \%100, i32 4, <8 x i1> \%86, <8 x float> undef), !tbaa !12,
                                                                                                                           ... !alias.scope !21
                                 %conv2.i.us.us = trunc i64 \sqrt[6]{116} to i32
                                                                                                                           %101 = fadd <8 x float> %96, %wide.masked.load63
                                %conv2.i.us.us = icmp sgt i32 %conv2.i.us.us, 0
%cmp11.i.us.us = icmp sgt i32 %sub.i, %conv2.i.us.us
%or.cond69.i.us.us = and i1 %cmp11.i.us.us, %cmp7.i.us.us
br i1 %or.cond69.i.us.us, label %if.then.i.us.us, label %if.end.i.us.us
                                                                                                                           %102 = add nsw i32 %mul26.i, %87
                                                                                                                           %103 = \text{sext i} 32 \% 102 \text{ to i} 64
                                                                                                                           %104 = getelementptr inbounds float, float* %0, i64 %103 %105 = bitcast float* %104 to <8 x float>*
                                                                                                                           %wide.masked.load64 = call <8 x float> @llvm.masked.load.v8f32.p0v8f32(<8 x
                                                                                                                           ... float>* %105, i32 4, <8 x i1> %86, <8 x float> undef), !tbaa !12,
                                                                                                                           ... !alias.scope !23
                                                                                                                           %106 = \text{fadd} < 8 \text{ x float} > %101, %wide.masked.load64
                                                                                                                           %107 = add nsw i32 %mul32.i, %87
                                                                                                                           %108 = \text{sext i} 32 \% 107 \text{ to i} 64
                                                                                                                           %109 = getelementptr inbounds float, float* %0, i64 %108 %110 = bitcast float* %109 to <8 x float>*
                                                                                                                           %wide.masked.load65 = call <8 x float> @llvm.masked.load.v8f32.p0v8f32(<8 x
                                                                                                                           ... float>* %110, i32 4, <8 x i1> %86, <8 x float> undef), !tbaa !12,
                                                                                                                           ... !alias.scope !25
                                                                                                                           %111 = \text{fadd} < 8 \text{ x float} > %106, %wide.masked.load65
                                                                                                                           %112 = \text{fmul} < 8 \text{ x float} > \%111, <float 0x3FC99999A0000000, float
                                                                                                                           ... 0x3FC99999A0000000, float 0x3FC99999A0000000, float 0x3FC99999A0000000, float
                                                                                                                           ... 0x3FC99999A0000000, float 0x3FC99999A0000000, float 0x3FC99999A0000000, float
                                                                                                                           ... 0x3FC99999A0000000>
                                                                                                                           %113 = getelementptr inbounds float, float* %1, i64 %89 %114 = bitcast float* %113 to <8 x float>*
                                                                                                                           call void @llvm.masked.store.v8f32.p0v8f32(<8 x float> %112, <8 x float>* ... %114, i32 4, <8 x i1> %86), !tbaa !12, !alias.scope !27, !noalias !29,
                                                                                                                           ...!llvm.access.group!30
                                                                                                                           %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>
                                                                                                                           %115 = icmp eq i64 %index.next, 32
                                                                                                                           br i1 %115, label %pregion_for_end.i.loopexit67, label %vector.body,
                                                                                                                           ...!llvm.loop!33
if.then.i.us.us:
%add.i.us.us = add i32 %mul.i, %conv2.i.us.us
%idxprom.i.us.us = sext i32 %add.i.us.us to i64
%arrayidx.i.us.us = getelementptr inbounds float, float* %0, i64
... %idxprom.i.us.us
%117 = load float, float* %arrayidx.i.us.us, align 4, !tbaa !12 %add15.i.us.us = add i32 %add.i.us.us, -1
%idxprom16.i.us.us = sext i32 %add15.i.us.us to i64
%arrayidx17.i.us.us = getelementptr inbounds float, float* %0, i64
... %idxprom16.i.us.us
%118 = load float, float* %arrayidx17.i.us.us, align 4, !tbaa !12
%add18.i.us.us = fadd float %117, %118
%add21.i.us.us = add i32 %add.i.us.us, 1
%idxprom22.i.us.us = sext i32 %add21.i.us.us to i64 %arrayidx23.i.us.us = getelementptr inbounds float, float* %0, i64
... %idxprom22.i.us.us
%119 = load float, float* %arrayidx23.i.us.us, align 4, !tbaa !12 %add24.i.us.us = fadd float %add18.i.us.us, %119 %add27.i.us.us = add nsw i32 %mul26.i, %conv2.i.us.us
%idxprom28.i.us.us = sext i32 %add27.i.us.us to i64
%arrayidx29.i.us.us = getelementptr inbounds float, float* %0, i64
... %idxprom28.i.us.us
%120 = load float, float* %arrayidx29.i.us.us, align 4, !tbaa !12
%add30.i.us.us = fadd float %add24.i.us.us, %120
%add33.i.us.us = add nsw i32 %mul32.i, %conv2.i.us.us
%idxprom34.i.us.us = sext i32 %add33.i.us.us to i64
%arrayidx35.i.us.us = getelementptr inbounds float, float* %0, i64
... %idxprom34.i.us.us
%121 = load float, float* %arrayidx35.i.us.us, align 4, !tbaa !12 %add36.i.us.us = fadd float %add30.i.us.us, %121
%mul37.i.us.us = fmul float %add36.i.us.us, 0x3FC99999A0000000
%arrayidx41.i.us.us = getelementptr inbounds float, float* %1, i64
... %idxprom.i.us.us
store float %mul37.i.us.us, float* %arrayidx41.i.us.us, align 4, !tbaa !12,
...!llvm.access.group!30
br label %if.end.i.us.us
                                                        if.end.i.us.us:
                                                        %122 = add nuw nsw i64 %_local_id_x.0.us.us, 1
                                                        %exitcond = icmp eq i64 %122, 32
br i1 %exitcond, label %pregion_for_end.i.loopexit, label
                                                                                                                                                                     pregion_for_end.i.loopexit67:
br label %pregion_for_end.i
                                                        ... %pregion for entry.entry.i.us.us, !Ilvm.loop !36
                                                                                                           pregion for end.i.loopexit:
                                                                                                            br label %pregion for end.i
                                                                                                                                                                          pregion for end.i:
                                                                                                                                                                          \sqrt{123} = ad\overline{d} nuw nsw i64 % local id v.0, 1
                                                                                                                                                                         %exitcond3 = icmp eq i64 %123, 8
br i1 %exitcond3, label %runJacobi2D_kernel1.exit, label
                                                                                                                                                                          ... %pregion for entry.pregion for init.i, !llvm.loop !37
                                                                                                                                                                                                                              F
                                                                                                                                                                          runJacobi2D_kernel1.exit:
                                                                                                                                                                          ret void
                                                                                                      CFG for 'pocl kernel runJacobi2D kernel1' function
```

%8 = shl i64 %5, 3