```
\%5 = tail call i64 @_Z13get_global_idj(i32 0) #3
                                                                              \%6 = \text{trunc } i64 \% 5 \text{ to } i32
                                                                              %7 = icmp slt i32 %6, %2
                                                                              br i1 %7, label %8, label %73
                                                                                                                   F
                                                                      %8:
                                                                      8:
                                                                      \%9 = \text{mul nsw i} 32 \%6, \%2
                                                                      %10 = icmp sgt i32 %3, 0
                                                                      %11 = \text{sext i} 32 \% 2 \text{ to i} 64
                                                                      %12 = \text{shl } i64 \%5, 32
                                                                      %13 = ashr exact i64 \%12, 32
                                                                      %14 = \text{sext i} 32 \% 9 \text{ to i} 64
                                                                      %15 = \text{zext i} 32 \% 3 \text{ to i} 64
                                                                      %16 = \text{ and } i64 \%15, 1
                                                                      %17 = icmp eq i32 %3, 1
                                                                      %18 = sub nsw i64 %15, %16
                                                                      %19 = icmp eq i64 %16, 0
                                                                      br label %20
                                                      %20:
                                                      20:
                                                      %21 = phi i64 [ %13, %8 ], [ %71, %65 ]
                                                       %22 = add nsw i64 %21, %14
                                                       %23 = getelementptr inbounds float, float* %0, i64 %22
                                                       store float 0.000000e+00, float* %23, align 4, !tbaa !10
                                                      br i1 %10, label %24, label %65
                                                                                                  F
                        %24:
                        24:
                        br i1 %17, label %49, label %25
                               Τ
                                                 F
            %25:
            25:
             \%26 = \text{phi float} [\%45, \%25], [0.000000e+00, \%24]
             %27 = phi i64 [ %46, %25 ], [ 0, %24 ]
             %28 = phi i64 [ %47, %25 ], [ %18, %24 ]
             %29 = mul nsw i64 %27, %11
             %30 = add nsw i64 %29, %13
             %31 = getelementptr inbounds float, float* %1, i64 %30
             %32 = load float, float* %31, align 4, !tbaa !10
             %33 = add nsw i64 %29, %21
             %34 = getelementptr inbounds float, float* %1, i64 %33
             %35 = load float, float* %34, align 4, !tbaa !10
             %36 = tail call float @llvm.fmuladd.f32(float %32, float %35, float %26)
             store float %36, float* %23, align 4, !tbaa !10
             %37 = \text{ or } i64 \%27, 1
             %38 = mul nsw i64 %37, %11
             %39 = add nsw i64 %38, %13
             %40 = getelementptr inbounds float, float* %1, i64 %39
             %41 = load float, float* %40, align 4, !tbaa !10
             %42 = add nsw i64 %38, %21
             %43 = getelementptr inbounds float, float* %1, i64 %42
             %44 = load float, float* %43, align 4, !tbaa !10
             %45 = tail call float @llvm.fmuladd.f32(float %41, float %44, float %36)
             store float %45, float* %23, align 4, !tbaa !10
             %46 = add nuw nsw i64 \%27, 2
             %47 = add i64 %28, -2
             %48 = icmp eq i64 \%47, 0
             br i1 %48, label %49, label %25
                                                                    F
%49:
49:
%50 = phi float [undef, %24], [%45, %25]
\%51 = \text{phi float} [0.0000000e+00, \%24], [\%45, \%25]
%52 = phi i64 [ 0, %24 ], [ %46, %25 ]
br i1 %19, label %62, label %53
               %53:
               53:
               %54 = mul nsw i64 %52, %11
               %55 = add nsw i64 %54, %13
               %56 = getelementptr inbounds float, float* %1, i64 %55
               %57 = load float, float* %56, align 4, !tbaa !10
               %58 = add nsw i64 %54, %21
               %59 = getelementptr inbounds float, float* %1, i64 %58
               %60 = load float, float* %59, align 4, !tbaa !10
               %61 = tail call float @llvm.fmuladd.f32(float %57, float %60, float %51)
               store float %61, float* %23, align 4, !tbaa !10
               br label %62
                              %62:
                              62:
                              %63 = phi float [ %50, %49 ], [ %61, %53 ]
                              \%64 = bitcast float \%63 to i32
                              br label %65
                                                   %65:
                                                   65:
                                                    %66 = phi i32 [ %64, %62 ], [ 0, %20 ]
                                                    %67 = mul nsw i64 %21, %11
                                                    %68 = add nsw i64 %67, %13
                                                    \%69 = \text{getelementptr inbounds float, float* } \%0, i64 \%68
                                                    \%70 = bitcast float* \%69 to i32*
                                                    store i32 %66, i32* %70, align 4, !tbaa !10
                                                    %71 = add nsw i64 %21, 1
                                                    \%72 = icmp \ eq \ i64 \ \%71, \ \%11
                                                    br i1 %72, label %73, label %20
                                                                Τ
                                                                                               F
                                                                                      %73:
                                                                                      73:
                                                                                      ret void
```

CFG for 'covar kernel' function