```
%7 = tail call i64 @ Z13get global idj(i32 0) #3
                                                         \%8 = \text{trunc } i64 \%7 \text{ to } i32
                                                          %9 = icmp slt i32 %8, %5
                                                         br i1 %9, label %10, label %68
                                                                                              F
                                                    %10:
                                                    10:
                                                     %11 = icmp sgt i32 \%5, 0
                                                     %12 = shl i64 \%7, 32
                                                     %13 = ashr exact i64 \%12, 32
                                                     br i1 %11, label %14, label %62
                                                           Τ
                                                                             F
 %14:
 14:
  %15 = getelementptr inbounds float, float* %1, i64 %13
  %16 = \text{zext i} 32 \% 5 \text{ to i} 64
  %17 = \text{shl } i64 \%7, 32
  %18 = ashr exact i64 \%17, 32
  %19 = \text{zext i} 32 \% 5 \text{ to i} 64
  %20 = load float, float* %15, align 4, !tbaa !10
  %21 = and i64 %19, 1
  %22 = icmp eq i32 %5, 1
  br i1 %22, label %49, label %23
              T
                                            F
                             %23:
                             23:
                              %24 = sub nsw i64 %19, %21
                              br label %25
        %25:
        25:
         %26 = phi float [ %20, %23 ], [ %45, %25 ]
         %27 = phi i64 [0, %23], [%46, %25]
         %28 = phi i64 [ %24, %23 ], [ %47, %25 ]
         %29 = mul nsw i64 %27, %16
         %30 = add nsw i64 %29, %18
         %31 = getelementptr inbounds float, float* %0, i64 %30
         %32 = Ĭoad float, float* %31, align 4, !tbaa !10
         %33 = fmul float %32, %4
         %34 = getelementptr inbounds float, float* %2, i64 %27
         %35 = load float, float* %34, align 4, !tbaa !10
         %36 = tail call float @llvm.fmuladd.f32(float %33, float %35, float %26)
         store float %36, float* %15, align 4, !tbaa !10
         %37 = \text{ or } i64 \%27, 1
         %38 = mul nsw i64 %37, %16
         %39 = add nsw i64 %38, %18
         %40 = getelementptr inbounds float, float* %0, i64 %39
         %41 = load float, float* %40, align 4, !tbaa !10
         %42 = fmul float %41, %4
         %43 = getelementptr inbounds float, float* %2, i64 %37
         %44 = load float, float* %43, align 4, !tbaa !10
         %45 = tail call float @llvm.fmuladd.f32(float %42, float %44, float %36)
         store float %45, float* %15, 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 [ %20, %14 ], [ %45, %25 ]
%51 = phi i64 [ 0, %14 ], [ %46, %25 ]
\%52 = icmp eq i64 \%21, 0
br i1 %52, label %62, label %53
           %53:
           53:
           %54 = mul nsw i64 %51, %16
           %55 = add nsw i64 %54, %18
           %56 = getelementptr inbounds float, float* %0, i64 %55
           %57 = load float, float* %56, align 4, !tbaa !10
           %58 = fmul float %57, %4
           %59 = getelementptr inbounds float, float* %2, i64 %51
           %60 = load float, float* %59, align 4, !tbaa !10
           %61 = tail call float @llvm.fmuladd.f32(float %58, float %60, float %50)
           store float %61, float* %15, align 4, !tbaa !10
           br label %62
                         %62:
                         62:
                          %63 = getelementptr inbounds float, float* %3, i64 %13
                          %64 = load float, float* %63, align 4, !tbaa !10
                          %65 = getelementptr inbounds float, float* %1, i64 %13
                          %66 = load float, float* %65, align 4, !tbaa !10
                          %67 = fadd float %64, %66
                          store float %67, float* %65, align 4, !tbaa !10
                          br label %68
                                                                   %68:
                                                                   68:
                                                                   ret void
```