```
%9 = tail call i64 @ Z13get global_idj(i32 0) #3
                                                                                %10 = \text{trunc } i64 \% 9 \text{ to } i32
                                                                                %11 = icmp slt i32 %10, %7
                                                                               br i1 %11, label %12, label %49
                                                                                                                     F
                                                                        %12:
                                                                        12:
                                                                         %13 = icmp sqt i32 \%7, 0
                                                                         br i1 %13, label %17, label %14
                                                                               Τ
                                                                                                 F
                     %17:
                     17:
                     %18 = mul nsw i32 %10, %7
                     %19 = \text{shl } i64 \%9, 32
                     %20 = ashr exact i64 %19, 32
                     %21 = getelementptr inbounds float, float* %4, i64 %20
                     %22 = getelementptr inbounds float, float* %3, i64 %20
                     %23 = \text{sext i} 32 \% 18 \text{ to i} 64
                     %24 = \text{zext i} 32 \% 7 \text{ to i} 64
                     br label %25
%25:
25:
\%26 = \text{phi } i64 [0, \%17], [\%39, \%25]
%27 = add nsw i64 %26, %23
%28 = getelementptr inbounds float, float* %0, i64 %27
%29 = load float, float* %28, align 4, !tbaa !10
%30 = getelementptr inbounds float, float* %2, i64 %26
%31 = load float, float* %30, align 4, !tbaa !10
%32 = load float, float* %21, align 4, !tbaa !10
                                                                                  %14:
%33 = tail call float @llvm.fmuladd.f32(float %29, float %31, float %32)
                                                                                  14:
store float %33, float* %21, align 4, !tbaa !10
                                                                                   %15 = \text{shl } i64 \%9, 32
%34 = getelementptr inbounds float, float* %1, i64 %27
                                                                                   %16 = ashr exact i64 \%15, 32
%35 = load float, float* %34, align 4, !tbaa !10
                                                                                   br label %41
%36 = load float, float* %30, align 4, !tbaa !10
%37 = load float, float* %22, align 4, !tbaa !10
%38 = tail call float @llvm.fmuladd.f32(float %35, float %36, float %37)
store float %38, float* %22, align 4, !tbaa !10
%39 = add nuw nsw i64 %26, 1
%40 = icmp eq i64 %39, %24
br i1 %40, label %41, label %25
                                                        F
                                        %41:
                                         %42 = phi i64 [ %16, %14 ], [ %20, %25 ]
                                         %43 = getelementptr inbounds float, float* %4, i64 %42
                                         %44 = load float, float* %43, align 4, !tbaa !10
                                         %45 = getelementptr inbounds float, float* %3, i64 %42
                                         %46 = load float, float* %45, align 4, !tbaa !10
                                         %47 = fmul float %46, %6
                                         %48 = tail call float @llvm.fmuladd.f32(float %5, float %44, float %47)
                                         store float %48, float* %45, align 4, !tbaa !10
                                         br label %49
                                                                                              %49:
                                                                                              49:
                                                                                               ret void
```