```
%6 = tail call i64 @ Z13get global idj(i32 0) #3
                                             \%7 = \text{trunc } i64 \%6 \text{ to } i32
                                             \%8 = icmp slt i32 \%7, \%3
                                             br i1 %8, label %9, label %55
                                                                                  F
                       %9:
                       9:
                       %10 = \text{shl } i64 \%6, 32
                       %11 = ashr exact i64 %10, 32
                       %12 = getelementptr inbounds float, float* %0, i64 %11
                       store float 0.000000e+00, float* %12, align 4, !tbaa !10
                       %13 = icmp sqt i32 %4, 0
                       br i1 %13, label %14, label %52
                                                                   F
                  %14:
                  14:
                   %15 = \text{sext i} 32 \% 3 \text{ to i} 64
                   %16 = shl i64 \%6, 32
                   %17 = ashr exact i64 \%16, 32
                   %18 = \text{zext i} 32 \% 4 \text{ to i} 64
                   %19 = \text{ and } i64 \%18, 1
                   %20 = icmp eq i32 %4, 1
                   br i1 %20, label %41, label %21
                         Τ
                                           F
                            %21:
                             %22 = sub nsw i64 %18, %19
                             br label %23
               %23:
               23:
               \%24 = \text{phi float} [ 0.0000000e+00, \%21 ], [ \%37, \%23 ]
               %25 = phi i64 [ 0, %21 ], [ %38, %23 ]
               %26 = phi i64 [ %22, %21 ], [ %39, %23 ]
               %27 = mul nsw i64 %25, %15
               %28 = add nsw i64 %27, %17
               %29 = getelementptr inbounds float, float* %1, i64 %28
               %30 = load float, float* %29, align 4, !tbaa !10
               %31 = fadd float %30, %24
               store float %31, float* %12, align 4, !tbaa !10
               %32 = \text{ or } i64 \%25, 1
               %33 = mul nsw i64 %32, %15
               %34 = add nsw i64 %33, %17
               %35 = getelementptr inbounds float, float* %1, i64 %34
               %36 = load float, float* %35, align 4, !tbaa !10
               %37 = fadd float %36, %31
               store float %37, float* %12, align 4, !tbaa !10
               \%38 = \text{add nuw nsw } i64 \%25, 2
               %39 = add i64 \%26, -2
               %40 = icmp eq i64 \%39, 0
               br i1 %40, label %41, label %23
                                                          F
%41:
41:
%42 = phi float [ undef, %14 ], [ %37, %23 ]
%43 = phi float [0.000000e+00, %14], [%37, %23]
%44 = phi i64 [ 0, %14 ], [ %38, %23 ]
%45 = icmp eq i64 \%19, 0
br i1 %45, label %52, label %46
                 %46:
                 46:
                  %47 = mul nsw i64 %44, %15
                  %48 = add nsw i64 %47, %17
                  %49 = getelementptr inbounds float, float* %1, i64 %48
                  %50 = load float, float* %49, align 4, !tbaa !10
                  %51 = fadd float %50, %43
                  store float %51, float* %12, align 4, !tbaa !10
                  br label %52
            %52:
            52:
             \%53 = \text{phi float} [0.000000e+00, \%9], [\%42, \%41], [\%51, \%46]
             %54 = fdiv float %53, %2, !fpmath !14
             store float %54, float* %12, align 4, !tbaa !10
             br label %55
                                                              %55:
                                                              55:
                                                               ret void
```

CFG for 'mean kernel' function

%5: