```
\%6 = \text{tail call i64} @ Z13\text{get global idj(i32 0) } #3
                          \%7 = \text{trunc } i64 \%6 \text{ to } i32
                          %8 = tail call i64 @ Z13get global idj(i32 1) #3
                          \%9 = \text{trunc } i64 \%8 \text{ to } i32
                          %10 = add nsw i32 %3, -1
                          %11 = icmp \ sqt \ i32 \ \%10, \ \%9
                          %12 = add nsw i32 %4, -1
                          %13 = icmp sgt i32 %12, %7
                          %14 = and i1 %13, %11
                          br i1 %14, label %15, label %42
                                     Т
                                                                F
%15:
15:
%16 = mul nsw i32 %9, %4
%17 = add i32 %16, %7
%18 = \text{sext i} 32 \%17 \text{ to i} 64
%19 = getelementptr inbounds float, float* %2, i64 %18
%20 = load float, float* %19, align 4, !tbaa !10
%21 = fpext float %20 to double
%22 = add i32 \%17, 1
%23 = \text{sext i} 32 \% 22 \text{ to i} 64
%24 = getelementptr inbounds float, float* %0, i64 %23
%25 = load float, float* %24, align 4, !tbaa !10
%26 = getelementptr inbounds float, float* %0, i64 %18
%27 = load float, float* %26, align 4, !tbaa !10
%28 = fsub float %25, %27
%29 = add \text{ nsw } i32 \%9, 1
%30 = \text{mul nsw i} 32 \% 29, \% 4
%31 = add nsw i32 %30, %7
%32 = \text{sext i} 32 \% 31 \text{ to i} 64
%33 = getelementptr inbounds float, float* %1, i64 %32
%34 = load float, float* %33, align 4, !tbaa !10
%35 = fadd float %28, %34
%36 = getelementptr inbounds float, float* %1, i64 %18
%37 = load float, float* %36, align 4, !tbaa !10
%38 = fsub float %35, %37
%39 = fpext float %38 to double
%40 = tail call double @llvm.fmuladd.f64(double %39, double
%41 = fptrunc double %40 to float
store float %41, float* %19, align 4, !tbaa !10
br label %42
                                               %42:
                                               42:
                                               ret void
```

CFG for 'fdtd kernel3' function

%5: