```
%6:
                                                     \%7 = tail call i64 @_Z13get_global_idj(i32 0) #3
                                                     \%8 = \text{trunc } i64 \%7 \text{ to } i32
                                                     %9 = tail call i64 @ Z13get global idj(i32 1) #3
                                                     %10 = \text{trunc } i64 \%9 \text{ to } i32
                                                     %11 = icmp slt i32 %10, %5
                                                     %12 = icmp slt i32 %8, %5
                                                     %13 = and i1 %12, %11
                                                     br i1 %13, label %14, label %69
                                                                Τ
                                                                                          F
                                %14:
                                14:
                                %15 = mul nsw i32 %10, %5
                                %16 = add nsw i32 %15, %8
                                %17 = \text{sext i} 32 \% 16 \text{ to i} 64
                                %18 = getelementptr inbounds float, float* %1, i64 %17
                                %19 = load float, float* %18, align 4, !tbaa !10
                                %20 = fmul float %19, %3
                                store float %20, float* %18, align 4, !tbaa !10
                                %21 = icmp sgt i32 %4, 0
                                br i1 %21, label %22, label %69
                                                                           F
                    %22:
                   22:
                    %23 = mul nsw i32 %10, %4
                    %24 = mul nsw i32 %8, %4
                    %25 = \text{sext i} 32 \% 24 \text{ to i} 64
                    %26 = \text{sext i} 32 \% 23 \text{ to i} 64
                    %27 = zext i32 \%4 to i64
                    %28 = and i64 %27, 1
                    %29 = icmp eq i32 %4, 1
                    br i1 %29, label %56, label %30
                          Τ
                                            F
                             %30:
                             30:
                              %31 = sub nsw i64 %27, %28
                              br label %32
        %32:
        32:
        %33 = phi float [ %20, %30 ], [ %52, %32 ]
         %34 = phi i64 [0, %30], [%53, %32]
         %35 = phi i64 [ %31, %30 ], [ %54, %32 ]
         %36 = add nsw i64 %34, %26
         %37 = getelementptr inbounds float, float* %0, i64 %36
         %38 = load float, float* %37, align 4, !tbaa !10
         %39 = fmul float %38, %2
         %40 = add nsw i64 %34, %25
         %41 = getelementptr inbounds float, float* %0, i64 %40
         %42 = Ĭoad float, float* %41, align 4, !tbaa !10
         %43 = tail call float @llvm.fmuladd.f32(float %39, float %42, float %33)
         store float %43, float* %18, align 4, !tbaa !10
         %44 = \text{ or } i64 \%34, 1
         %45 = add nsw i64 %44, %26
         %46 = getelementptr inbounds float, float* %0, i64 %45
         %47 = load float, float* %46, align 4, !tbaa !10
         %48 = fmul float %47, %2
         %49 = add nsw i64 %44, %25
         %50 = getelementptr inbounds float, float* %0, i64 %49
         %51 = load float, float* %50, align 4, !tbaa !10
         %52 = tail call float @llvm.fmuladd.f32(float %48, float %51, float %43)
         store float %52, float* %18, align 4, !tbaa !10
         %53 = add nuw nsw i64 %34, 2
         %54 = add i64 %35, -2
         %55 = icmp eq i64 \%54, 0
         br i1 %55, label %56, label %32
                                                               F
%56:
56:
%57 = phi float [ %20, %22 ], [ %52, %32 ]
%58 = phi i64 [ 0, %22 ], [ %53, %32 ]
%59 = icmp eq i64 %28, 0
br i1 %59, label %69, label %60
         Т
                                 F
           %60:
           60:
           %61 = add nsw i64 %58, %26
           %62 = getelementptr inbounds float, float* %0, i64 %61
           %63 = load float, float* %62, align 4, !tbaa !10
           %64 = fmul float %63, %2
           %65 = add nsw i64 %58, %25
           %66 = getelementptr inbounds float, float* %0, i64 %65
           %67 = load float, float* %66, align 4, !tbaa !10
           %68 = tail call float @llvm.fmuladd.f32(float %64, float %67, float %57)
           store float %68, float* %18, align 4, !tbaa !10
           br label %69
                                                                %69:
                                                                69:
                                                                 ret void
```

CFG for 'syrk kernel' function