

%5:  
%6 = icmp eq float\* %1, null  
%7 = icmp eq float\* %2, null  
%8 = select i1 %6, i1 true, i1 %7  
%9 = icmp eq i8\* %3, null  
%10 = select i1 %8, i1 true, i1 %9  
%11 = icmp eq float\* %4, null  
%12 = select i1 %10, i1 true, i1 %11  
br i1 %12, label %100, label %13

T

F

%13:  
13:  
%14 = fpext float %0 to double  
%15 = call double @llvm.cos.f64(double %14)  
%16 = fptrunc double %15 to float  
%17 = call double @llvm.sin.f64(double %14)  
%18 = fptrunc double %17 to float  
%19 = fadd float %16, 2.000000e+00  
br label %20

%20:  
20:  
%21 = phi double [ 0.000000e+00, %13 ], [ %98, %95 ]  
%22 = call double @llvm.sin.f64(double %21)  
%23 = fptrunc double %22 to float  
%24 = load float, float\* %1, align 4, !tbaa !10  
%25 = fpext float %24 to double  
%26 = call double @llvm.sin.f64(double %25)  
%27 = fptrunc double %26 to float  
%28 = call double @llvm.cos.f64(double %25)  
%29 = fptrunc double %28 to float  
%30 = fmul float %19, %23  
%31 = fmul float %30, %27  
%32 = fmul float %18, %29  
%33 = fadd float %31, %32  
%34 = fadd float %33, 5.000000e+00  
%35 = fdiv float 1.000000e+00, %34  
%36 = call double @llvm.cos.f64(double %21)  
%37 = fptrunc double %36 to float  
%38 = load float, float\* %2, align 4, !tbaa !10  
%39 = fpext float %38 to double  
%40 = call double @llvm.cos.f64(double %39)  
%41 = fptrunc double %40 to float  
%42 = call double @llvm.sin.f64(double %39)  
%43 = fptrunc double %42 to float  
%44 = fmul float %30, %29  
%45 = fmul float %18, %27  
%46 = fsub float %44, %45  
%47 = fmul float %35, 3.000000e+01  
%48 = fmul float %19, %37  
%49 = fmul float %48, %41  
%50 = fmul float %46, %43  
%51 = fsub float %49, %50  
%52 = fmul float %51, %47  
%53 = fadd float %52, 4.000000e+01  
%54 = fptosi float %53 to i32  
%55 = fmul float %35, 1.500000e+01  
%56 = fmul float %48, %43  
%57 = fmul float %46, %41  
%58 = fadd float %56, %57  
%59 = fmul float %58, %55  
%60 = fadd float %59, 1.200000e+01  
%61 = fptosi float %60 to i32  
%62 = fmul float %16, %23  
%63 = fmul float %62, %29  
%64 = fsub float %45, %63  
%65 = fmul float %64, %41  
%66 = fmul float %62, %27  
%67 = fsub float %65, %66  
%68 = fsub float %67, %32  
%69 = fmul float %16, %37  
%70 = fmul float %69, %43  
%71 = fsub float %68, %70  
%72 = fmul float %71, 8.000000e+00  
%73 = fptosi float %72 to i32  
%74 = add i32 %61, -1  
%75 = icmp ult i32 %74, 21  
%76 = icmp sgt i32 %54, 0  
%77 = select i1 %75, i1 %76, i1 false  
%78 = icmp slt i32 %54, 80  
%79 = select i1 %77, i1 %78, i1 false  
br i1 %79, label %80, label %95

T

F

%80:  
80:  
%81 = mul nsw i32 %61, 80  
%82 = add nsw i32 %81, %54  
%83 = sext i32 %82 to i64  
%84 = getelementptr inbounds float, float\* %4, i64 %83  
%85 = load float, float\* %84, align 4, !tbaa !10  
%86 = fcmp ogt float %35, %85  
br i1 %86, label %87, label %95

T

F

%87:  
87:  
store float %35, float\* %84, align 4, !tbaa !10  
%88 = load i8\*, i8\*\* @CHARS, align 8, !tbaa !14  
%89 = icmp sgt i32 %73, 0  
%90 = select i1 %89, i32 %73, i32 0  
%91 = zext i32 %90 to i64  
%92 = getelementptr inbounds i8, i8\* %88, i64 %91  
%93 = load i8, i8\* %92, align 1, !tbaa !16  
%94 = getelementptr inbounds i8, i8\* %3, i64 %83  
store i8 %93, i8\* %94, align 1, !tbaa !16  
br label %95

%95:  
95:  
%96 = fadd double %21, 2.000000e-02  
%97 = fptrunc double %96 to float  
%98 = fpext float %97 to double  
%99 = fcmp olt double %98, 6.280000e+00  
br i1 %99, label %20, label %100, !llvm.loop !17

T

F

%100:  
100:  
%101 = phi i32 [ 2, %5 ], [ 0, %95 ]  
ret i32 %101

CFG for 'inner\_loop' function