

0:

```
%pgocount4 = load i64, ptr getelementptr inbounds ([4 x i64], ptr
... @_profc_main.2, i32 0, i32 3), align 8
%1 = add i64 %pgocount4, 1
store i64 %1, ptr getelementptr inbounds ([4 x i64], ptr @_profc_main.2,
... i32 0, i32 3), align 8
%pgocount = load i64, ptr getelementptr inbounds ([4 x i64], ptr
... @_profc_main, i32 0, i32 3), align 8
%2 = add i64 %pgocount, 1
store i64 %2, ptr getelementptr inbounds ([4 x i64], ptr @_profc_main, i32
... 0, i32 3), align 8
%3 = alloca i32, align 4
%4 = alloca [10 x i32], align 16
%5 = alloca [10 x i32], align 16
%6 = alloca i32, align 4
%7 = alloca i32, align 4
%8 = alloca i32, align 4
store i32 0, ptr %3, align 4
call void @llvm.memcpy.p0.p0.i64(ptr align 16 %4, ptr align 16
... @_const.main.A, i64 40, i1 false)
call void @llvm.memset.p0.i64(ptr align 16 %5, i8 0, i64 40, i1 false)
store i32 37, ptr %8, align 4
store i32 0, ptr %7, align 4
store i32 0, ptr %6, align 4
%newreg_load_src = alloca i32, align 4
%9 = load i32, ptr %8, align 4
store i32 %9, ptr %newreg_load_src, align 4
%newreg_load_src1 = alloca i32, align 4
%10 = load i32, ptr %7, align 4
store i32 %10, ptr %newreg_load_src1, align 4
br label %11
```

11:

```
%12 = load i32, ptr %6, align 4
%13 = icmp slt i32 %12, 10
br i1 %13, label %14, label %54, !prof !36
```

T

F

14:

```
%newreg_load_dst = load i32, ptr %newreg_load_src, align 4
%15 = mul nsw i32 %newreg_load_dst, 2
%newreg_load_dst2 = load i32, ptr %newreg_load_src1, align 4
%16 = sext i32 %newreg_load_dst2 to i64
%17 = getelementptr inbounds [10 x i32], ptr %4, i64 0, i64 %16
%18 = load i32, ptr %17, align 4
%19 = mul nsw i32 %18, 23
%20 = add nsw i32 %15, %19
%21 = load i32, ptr %6, align 4
%22 = add nsw i32 %20, %21
%23 = load i32, ptr %6, align 4
%24 = sext i32 %23 to i64
%25 = getelementptr inbounds [10 x i32], ptr %5, i64 0, i64 %24
store i32 %22, ptr %25, align 4
%26 = load i32, ptr %6, align 4
%27 = srem i32 %26, 7
%28 = icmp eq i32 %27, 0
br i1 %28, label %29, label %43, !prof !37
```

T

F

54:

```
ret i32 0
```

29:

```
%30 = load i32, ptr %6, align 4
%31 = srem i32 %30, 2
%32 = icmp eq i32 %31, 1
br i1 %32, label %33, label %37, !prof !38
```

T

F

33:

```
%pgocount5 = load i64, ptr getelementptr inbounds ([4 x i64], ptr
... @_profc_main.2, i32 0, i32 1), align 8
%34 = add i64 %pgocount5, 1
store i64 %34, ptr getelementptr inbounds ([4 x i64], ptr @_profc_main.2,
... i32 0, i32 1), align 8
%pgocount1 = load i64, ptr getelementptr inbounds ([4 x i64], ptr
... @_profc_main, i32 0, i32 1), align 8
%35 = add i64 %pgocount1, 1
store i64 %35, ptr getelementptr inbounds ([4 x i64], ptr @_profc_main, i32
... 0, i32 1), align 8
%36 = load i32, ptr %6, align 4
store i32 %36, ptr %newreg_load_src1, align 4
br label %42
```

37:

```
%pgocount6 = load i64, ptr getelementptr inbounds ([4 x i64], ptr
... @_profc_main.2, i32 0, i32 2), align 8
%38 = add i64 %pgocount6, 1
store i64 %38, ptr getelementptr inbounds ([4 x i64], ptr @_profc_main.2,
... i32 0, i32 2), align 8
%pgocount2 = load i64, ptr getelementptr inbounds ([4 x i64], ptr
... @_profc_main, i32 0, i32 2), align 8
%39 = add i64 %pgocount2, 1
store i64 %39, ptr getelementptr inbounds ([4 x i64], ptr @_profc_main, i32
... 0, i32 2), align 8
%40 = load i32, ptr %6, align 4
%41 = add nsw i32 %40, 1
store i32 %41, ptr %newreg_load_src, align 4
br label %42
```

42:

```
br label %43
```

43:

```
%44 = load i32, ptr %6, align 4
%45 = sext i32 %44 to i64
%46 = getelementptr inbounds [10 x i32], ptr %5, i64 0, i64 %45
%47 = load i32, ptr %46, align 4
%48 = call i32 (ptr, ...) @printf(ptr noundef @.str, i32 noundef %47)
br label %49
```

49:

```
%pgocount7 = load i64, ptr @_profc_main.2, align 8
%50 = add i64 %pgocount7, 1
store i64 %50, ptr @_profc_main.2, align 8
%pgocount3 = load i64, ptr @_profc_main, align 8
%51 = add i64 %pgocount3, 1
store i64 %51, ptr @_profc_main, align 8
%52 = load i32, ptr %6, align 4
%53 = add nsw i32 %52, 1
store i32 %53, ptr %6, align 4
br label %11, !llvm.loop !39
```

CFG for 'main' function