

```

%0:
%pgocount = load i64, ptr getelementptr inbounds ([2 x i64], ptr
... @_profc_main, i32 0, i32 1), align 8
%1 = add i64 %pgocount, 1
store i64 %1, ptr getelementptr inbounds ([2 x i64], ptr @_profc_main, i32
... 0, i32 1), align 8
%2 = alloca i32, align 4
%3 = alloca [10 x i32], align 16
%4 = alloca i32, align 4
store i32 0, ptr %2, align 4
call void @llvm.memset.p0.i64(ptr align 16 %3, i8 0, i64 40, i1 false)
store i32 0, ptr %4, align 4
br label %5

```

```

%5:
5:
%6 = load i32, ptr %4, align 4
%7 = icmp slt i32 %6, 11
br i1 %7, label %8, label %20, !prof !36

```

T	F
---	---

```

%8:
8:
%9 = load i32, ptr %4, align 4
%10 = sext i32 %9 to i64
%11 = icmp slt i64 %10, 0
br i1 %11, label %ifBlock, label %split

```

T	F
---	---

```

ifBlock:
%21 = call i32 (ptr, ...) @printf(ptr @str)
call void @exit(i32 1)
br label %split

```

```

split:
%12 = icmp slt i64 9, %10
br i1 %12, label %ifBlock2, label %split1

```

T	F
---	---

```

ifBlock2:
%22 = call i32 (ptr, ...) @printf(ptr @str.1)
call void @exit(i32 1)
br label %split1

```

```

split1:
%13 = getelementptr inbounds [10 x i32], ptr %3, i64 0, i64 %10
%14 = load i32, ptr %13, align 4
%15 = call i32 (ptr, ...) @printf(ptr noundef @.str, i32 noundef %14)
br label %16

```

```

%16:
16:
%pgocount1 = load i64, ptr @_profc_main, align 8
%17 = add i64 %pgocount1, 1
store i64 %17, ptr @_profc_main, align 8
%18 = load i32, ptr %4, align 4
%19 = add nsw i32 %18, 1
store i32 %19, ptr %4, align 4
br label %5, !llvm.loop !37

```

```

%20:
20:
ret i32 0

```

CFG for 'main' function