

```
%0:
%1 = alloca i32, align 4
%2 = alloca [10 x i32], align 16
%3 = alloca [20 x i32], align 16
%4 = alloca [5 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 %1, align 4
call void @llvm.memset.p0.i64(ptr align 16 %2, i8 0, i64 40, i1 false)
call void @llvm.memset.p0.i64(ptr align 16 %3, i8 0, i64 80, i1 false)
call void @llvm.memset.p0.i64(ptr align 16 %4, i8 0, i64 20, i1 false)
call void @llvm.memset.p0.i64(ptr align 16 %5, i8 0, i64 40, i1 false)
store i32 0, ptr %6, align 4
br label %9
```

```
%9:
9:
%10 = load i32, ptr %6, align 4
%11 = icmp slt i32 %10, 10
br i1 %11, label %12, label %134
```

```
%12:
12:
store i32 0, ptr %7, align 4
br label %13
```

```
%134:
134:
ret i32 0
```

```
%13:
13:
%14 = load i32, ptr %7, align 4
%15 = icmp slt i32 %14, 10
br i1 %15, label %16, label %130
```

```
%130:
130:
br label %131
```

```
%131:
131:
%132 = load i32, ptr %6, align 4
%133 = add nsw i32 %132, 1
store i32 %133, ptr %6, align 4
br label %9, !llvm.loop !9
```

```
%16:
16:
store i32 0, ptr %8, align 4
br label %17
```

```
%17:
17:
%18 = load i32, ptr %8, align 4
%19 = icmp slt i32 %18, 10
br i1 %19, label %20, label %126
```

```
%126:
126:
br label %127
```

```
%127:
127:
%128 = load i32, ptr %7, align 4
%129 = add nsw i32 %128, 1
store i32 %129, ptr %7, align 4
br label %13, !llvm.loop !8
```

```
%20:
20:
%21 = load i32, ptr %6, align 4
%22 = sub nsw i32 %21, 0
%23 = sext i32 %22 to i64
%24 = icmp slt i64 %23, 0
br i1 %24, label %ifBlock, label %split
```

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

```
split:
%25 = icmp slt i64 9, %23
br i1 %25, label %ifBlock2, label %split1
```

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

```
split1:
%26 = getelementptr inbounds [10 x i32], ptr %2, i64 0, i64 %23
%27 = load i32, ptr %26, align 4
%28 = call i32 (ptr, ...) @printf(ptr noundef @.str, i32 noundef %27)
%29 = load i32, ptr %6, align 4
%30 = sub nsw i32 %29, 0
%31 = mul nsw i32 %30, 2
%32 = sext i32 %31 to i64
%33 = icmp slt i64 %32, 0
br i1 %33, label %ifBlock4, label %split3
```

```
ifBlock4:
%137 = call i32 (ptr, ...) @printf(ptr @str.2)
call void @exit(i32 1)
br label %split3
```

```
split3:
%34 = icmp slt i64 19, %32
br i1 %34, label %ifBlock6, label %split5
```

```
ifBlock6:
%138 = call i32 (ptr, ...) @printf(ptr @str.3)
call void @exit(i32 1)
br label %split5
```

```
split5:
%35 = getelementptr inbounds [20 x i32], ptr %3, i64 0, i64 %32
%36 = load i32, ptr %35, align 4
%37 = call i32 (ptr, ...) @printf(ptr noundef @.str, i32 noundef %36)
%38 = load i32, ptr %6, align 4
%39 = sub nsw i32 %38, 0
%40 = sdiv i32 %39, 2
%41 = sext i32 %40 to i64
%42 = icmp slt i64 %41, 0
br i1 %42, label %ifBlock8, label %split7
```

```
ifBlock8:
%139 = call i32 (ptr, ...) @printf(ptr @str.4)
call void @exit(i32 1)
br label %split7
```

```
split7:
%43 = icmp slt i64 4, %41
br i1 %43, label %ifBlock10, label %split9
```

```
ifBlock10:
%140 = call i32 (ptr, ...) @printf(ptr @str.5)
call void @exit(i32 1)
br label %split9
```

```
split9:
%44 = getelementptr inbounds [5 x i32], ptr %4, i64 0, i64 %41
%45 = load i32, ptr %44, align 4
%46 = call i32 (ptr, ...) @printf(ptr noundef @.str, i32 noundef %45)
%47 = load i32, ptr %6, align 4
%48 = sub nsw i32 %47, 0
%49 = sext i32 %48 to i64
%50 = icmp slt i64 %49, 0
br i1 %50, label %ifBlock12, label %split11
```

```
ifBlock12:
%141 = call i32 (ptr, ...) @printf(ptr @str.6)
call void @exit(i32 1)
br label %split11
```

```
split11:
%51 = icmp slt i64 9, %49
br i1 %51, label %ifBlock14, label %split13
```

```
ifBlock14:
%142 = call i32 (ptr, ...) @printf(ptr @str.7)
call void @exit(i32 1)
br label %split13
```

```
split13:
%52 = getelementptr inbounds [10 x i32], ptr %5, i64 0, i64 %49
%53 = load i32, ptr %52, align 4
%54 = call i32 (ptr, ...) @printf(ptr noundef @.str, i32 noundef %53)
%55 = load i32, ptr %7, align 4
%56 = sub nsw i32 %55, 0
%57 = sext i32 %56 to i64
%58 = icmp slt i64 %57, 0
br i1 %58, label %ifBlock16, label %split15
```

```
ifBlock16:
%143 = call i32 (ptr, ...) @printf(ptr @str.8)
call void @exit(i32 1)
br label %split15
```

```
split15:
%59 = icmp slt i64 9, %57
br i1 %59, label %ifBlock18, label %split17
```

```
ifBlock18:
%144 = call i32 (ptr, ...) @printf(ptr @str.9)
call void @exit(i32 1)
br label %split17
```

```
split17:
%60 = getelementptr inbounds [10 x i32], ptr %2, i64 0, i64 %57
%61 = load i32, ptr %60, align 4
%62 = call i32 (ptr, ...) @printf(ptr noundef @.str, i32 noundef %61)
%63 = load i32, ptr %7, align 4
%64 = sub nsw i32 %63, 0
%65 = mul nsw i32 %64, 2
%66 = sext i32 %65 to i64
%67 = icmp slt i64 %66, 0
br i1 %67, label %ifBlock20, label %split19
```

```
ifBlock20:
%145 = call i32 (ptr, ...) @printf(ptr @str.10)
call void @exit(i32 1)
br label %split19
```

```
split19:
%68 = icmp slt i64 19, %66
br i1 %68, label %ifBlock22, label %split21
```

```
ifBlock22:
%146 = call i32 (ptr, ...) @printf(ptr @str.11)
call void @exit(i32 1)
br label %split21
```

```
split21:
%69 = getelementptr inbounds [20 x i32], ptr %3, i64 0, i64 %66
%70 = load i32, ptr %69, align 4
%71 = call i32 (ptr, ...) @printf(ptr noundef @.str, i32 noundef %70)
%72 = load i32, ptr %7, align 4
%73 = sub nsw i32 %72, 0
%74 = sdiv i32 %73, 2
%75 = sext i32 %74 to i64
%76 = icmp slt i64 %75, 0
br i1 %76, label %ifBlock24, label %split23
```

```
ifBlock24:
%147 = call i32 (ptr, ...) @printf(ptr @str.12)
call void @exit(i32 1)
br label %split23
```

```
split23:
%77 = icmp slt i64 4, %75
br i1 %77, label %ifBlock26, label %split25
```

```
ifBlock26:
%148 = call i32 (ptr, ...) @printf(ptr @str.13)
call void @exit(i32 1)
br label %split25
```

```
split25:
%78 = getelementptr inbounds [5 x i32], ptr %4, i64 0, i64 %75
%79 = load i32, ptr %78, align 4
%80 = call i32 (ptr, ...) @printf(ptr noundef @.str, i32 noundef %79)
%81 = load i32, ptr %7, align 4
%82 = sub nsw i32 %81, 0
%83 = sext i32 %82 to i64
%84 = icmp slt i64 %83, 0
br i1 %84, label %ifBlock28, label %split27
```

```
ifBlock28:
%149 = call i32 (ptr, ...) @printf(ptr @str.14)
call void @exit(i32 1)
br label %split27
```

```
split27:
%85 = icmp slt i64 9, %83
br i1 %85, label %ifBlock30, label %split29
```

```
ifBlock30:
%150 = call i32 (ptr, ...) @printf(ptr @str.15)
call void @exit(i32 1)
br label %split29
```

```
split29:
%86 = getelementptr inbounds [10 x i32], ptr %5, i64 0, i64 %83
%87 = load i32, ptr %86, align 4
%88 = call i32 (ptr, ...) @printf(ptr noundef @.str, i32 noundef %87)
%89 = load i32, ptr %8, align 4
%90 = sub nsw i32 %89, 0
%91 = sext i32 %90 to i64
%92 = icmp slt i64 %91, 0
br i1 %92, label %ifBlock32, label %split31
```

```
ifBlock32:
%151 = call i32 (ptr, ...) @printf(ptr @str.16)
call void @exit(i32 1)
br label %split31
```

```
split31:
%93 = icmp slt i64 9, %91
br i1 %93, label %ifBlock34, label %split33
```

```
ifBlock34:
%152 = call i32 (ptr, ...) @printf(ptr @str.17)
call void @exit(i32 1)
br label %split33
```

```
split33:
%94 = getelementptr inbounds [10 x i32], ptr %2, i64 0, i64 %91
%95 = load i32, ptr %94, align 4
%96 = call i32 (ptr, ...) @printf(ptr noundef @.str, i32 noundef %95)
%97 = load i32, ptr %8, align 4
%98 = sub nsw i32 %97, 0
%99 = mul nsw i32 %98, 2
%100 = sext i32 %99 to i64
%101 = icmp slt i64 %100, 0
br i1 %101, label %ifBlock36, label %split35
```

```
ifBlock36:
%153 = call i32 (ptr, ...) @printf(ptr @str.18)
call void @exit(i32 1)
br label %split35
```

```
split35:
%102 = icmp slt i64 19, %100
br i1 %102, label %ifBlock38, label %split37
```

```
ifBlock38:
%154 = call i32 (ptr, ...) @printf(ptr @str.19)
call void @exit(i32 1)
br label %split37
```

```
split37:
%103 = getelementptr inbounds [20 x i32], ptr %3, i64 0, i64 %100
%104 = load i32, ptr %103, align 4
%105 = call i32 (ptr, ...) @printf(ptr noundef @.str, i32 noundef %104)
%106 = load i32, ptr %8, align 4
%107 = sub nsw i32 %106, 0
%108 = sdiv i32 %107, 2
%109 = sext i32 %108 to i64
%110 = icmp slt i64 %109, 0
br i1 %110, label %ifBlock40, label %split39
```

```
ifBlock40:
%155 = call i32 (ptr, ...) @printf(ptr @str.20)
call void @exit(i32 1)
br label %split39
```

```
split39:
%111 = icmp slt i64 4, %109
br i1 %111, label %ifBlock42, label %split41
```

```
ifBlock42:
%156 = call i32 (ptr, ...) @printf(ptr @str.21)
call void @exit(i32 1)
br label %split41
```

```
split41:
%112 = getelementptr inbounds [5 x i32], ptr %4, i64 0, i64 %109
%113 = load i32, ptr %112, align 4
%114 = call i32 (ptr, ...) @printf(ptr noundef @.str, i32 noundef %113)
%115 = load i32, ptr %8, align 4
%116 = sub nsw i32 %115, 0
%117 = sext i32 %116 to i64
%118 = icmp slt i64 %117, 0
br i1 %118, label %ifBlock44, label %split43
```

```
ifBlock44:
%157 = call i32 (ptr, ...) @printf(ptr @str.22)
call void @exit(i32 1)
br label %split43
```

```
split43:
%119 = icmp slt i64 9, %117
br i1 %119, label %ifBlock46, label %split45
```

```
ifBlock46:
%158 = call i32 (ptr, ...) @printf(ptr @str.23)
call void @exit(i32 1)
br label %split45
```

```
split45:
%120 = getelementptr inbounds [10 x i32], ptr %5, i64 0, i64 %117
%121 = load i32, ptr %120, align 4
%122 = call i32 (ptr, ...) @printf(ptr noundef @.str, i32 noundef %121)
br label %123
```

```
%123:
123:
%124 = load i32, ptr %8, align 4
%125 = add nsw i32 %124, 1
store i32 %125, ptr %8, align 4
br label %17, !llvm.loop !6
```