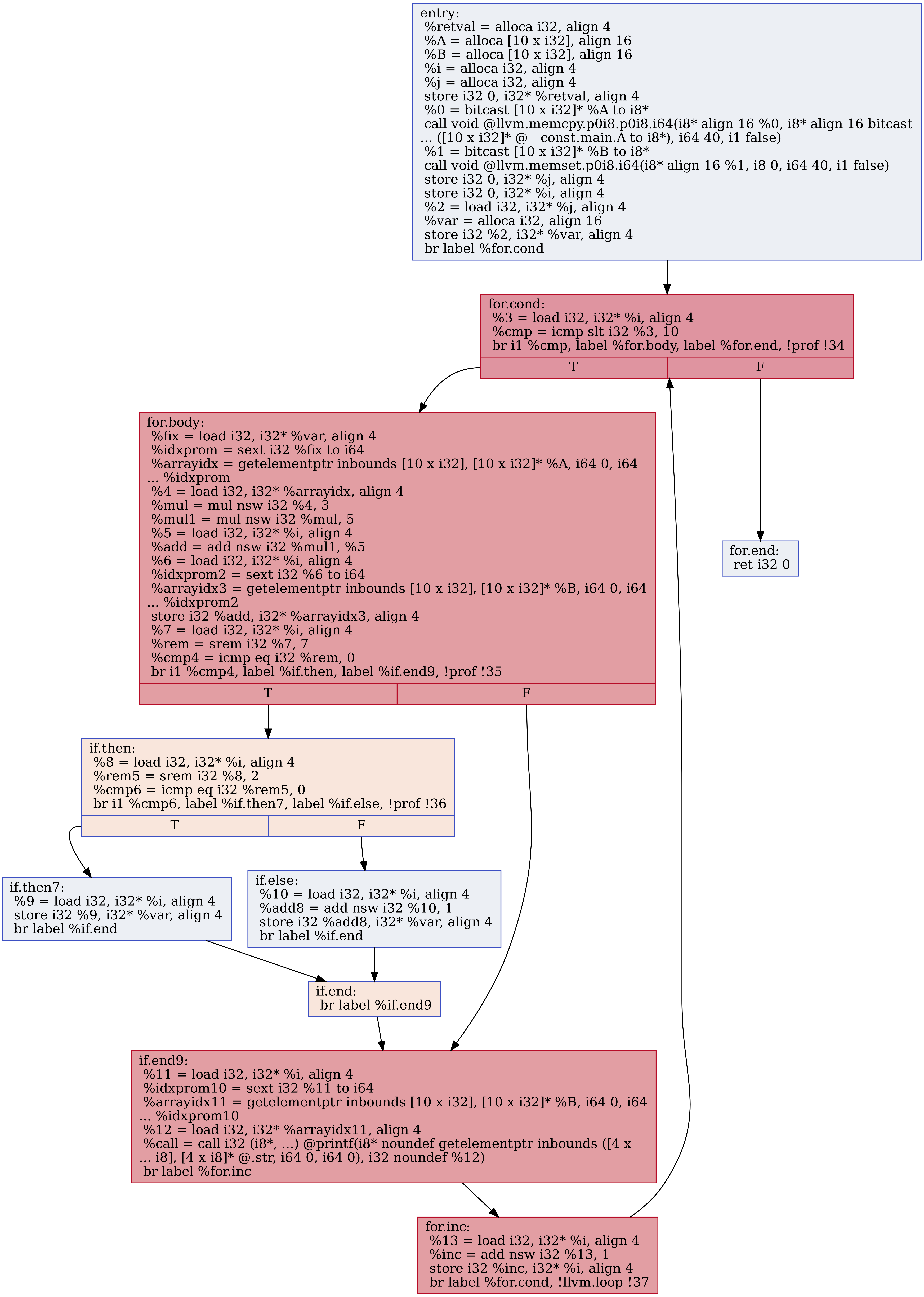

```
entry:
 %retval = alloca i32, align 4
 %A = alloca [10 x i32], align 16
 %B = alloca [10 x i32], align 16
 %i = alloca i32, align 4
 %j = alloca i32, align 4
 store i32 0, i32* %retval, align 4
 %0 = bitcast [10 x i32]* %A to i8*
 call void @llvm.memcpy.p0i8.p0i8.i64(i8* align 16 %0, i8* align 16 bitcast
... ([10 x i32]* @__const.main.A to i8*), i64 40, i1 false)
 %1 = bitcast [10 x i32]* %B to i8*
 call void @llvm.memset.p0i8.i64(i8* align 16 %1, i8 0, i64 40, i1 false)
 store i32 0, i32* %j, align 4
 store i32 0, i32* %i, align 4
 %2 = load i32, i32* %j, align 4
 %var = alloca i32, align 16
 store i32 %2, i32* %var, align 4
 br label %for.cond
```

```
for.cond:
 %3 = load i32, i32* %i, align 4
 %cmp = icmp slt i32 %3, 10
 br i1 %cmp, label %for.body, label %for.end, !prof !34
```

```
for.body:
 %fix = load i32, i32* %var, align 4
 %idxprom = sext i32 %fix to i64
 %arrayidx = getelementptr inbounds [10 x i32], [10 x i32]* %A, i64 0, i64
... %idxprom
 %4 = load i32, i32* %arrayidx, align 4
 %mul = mul nsw i32 %4, 3
 %mul1 = mul nsw i32 %mul, 5
 %5 = load i32, i32* %i, align 4
 %add = add nsw i32 %mul1, %5
 %6 = load i32, i32* %i, align 4
 %idxprom2 = sext i32 %6 to i64
 %arrayidx3 = getelementptr inbounds [10 x i32], [10 x i32]* %B, i64 0, i64
... %idxprom2
 store i32 %add, i32* %arrayidx3, align 4
 %7 = load i32, i32* %i, align 4
 %rem = srem i32 %7, 7
 %cmp4 = icmp eq i32 %rem, 0
 br i1 %cmp4, label %if.then, label %if.end9, !prof !35
```

```
for.end:
 ret i32 0
```

```
if.then:
 %8 = load i32, i32* %i, align 4
 %rem5 = srem i32 %8, 2
 %cmp6 = icmp eq i32 %rem5, 0
 br i1 %cmp6, label %if.then7, label %if.else, !prof !36
```

```
if.then7:
 %9 = load i32, i32* %i, align 4
 store i32 %9, i32* %var, align 4
 br label %if.end
```

```
if.else:
 %10 = load i32, i32* %i, align 4
 %add8 = add nsw i32 %10, 1
 store i32 %add8, i32* %var, align 4
 br label %if.end
```

```
if.end:
 br label %if.end9
```

```
if.end9:
 %11 = load i32, i32* %i, align 4
 %idxprom10 = sext i32 %11 to i64
 %arrayidx11 = getelementptr inbounds [10 x i32], [10 x i32]* %B, i64 0, i64
... %idxprom10
 %12 = load i32, i32* %arrayidx11, align 4
 %call = call i32 (i8*, ...) @printf(i8* noundef getelementptr inbounds ([4 x
... i8], [4 x i8]* @.str, i64 0, i64 0), i32 noundef %12)
 br label %for.inc
```

```
for.inc:
 %13 = load i32, i32* %i, align 4
 %inc = add nsw i32 %13, 1
 store i32 %inc, i32* %i, align 4
 br label %for.cond, !llvm.loop !37
```

CFG for 'main' function