

[8/11]  
for.cond: ; preds = %for.inc, %entry  
%0 = load i32, i32\* %i, align 4  
%cmp = icmp ult i32 %0, 100  
br i1 %cmp, label %for.body, label %for.end



[9/10]  
for.body: ; preds = %for.cond  
%call1 = call i32 @rand() #2  
%rem = urem i32 %call1, 10  
%1 = load i32, i32\* %i, align 4  
%idxprom = zext i32 %1 to i64  
%arrayidx = getelementptr inbounds [100 x i32], [100 x i32]\* %a, i64 0, i64 %idxprom  
store i32 %rem, i32\* %arrayidx, align 4  
%call2 = call i32 @rand() #2  
%rem3 = urem i32 %call2, 10  
%2 = load i32, i32\* %i, align 4  
%idxprom4 = zext i32 %2 to i64  
%arrayidx5 = getelementptr inbounds [100 x i32], [100 x i32]\* %b, i64 0, i64 %idxprom4  
store i32 %rem3, i32\* %arrayidx5, align 4  
br label %for.inc

[1/2]  
for.inc20: ; preds = %for.body9  
%13 = load i32, i32\* %i6, align 4  
%inc21 = add i32 %13, 1  
store i32 %inc21, i32\* %i6, align 4  
br label %for.cond7

[0/5]  
for.cond7: ; preds = %for.inc20, %for.end  
%4 = load i32, i32\* %i6, align 4  
%cmp8 = icmp ult i32 %4, 100  
br i1 %cmp8, label %for.body9, label %for.end22

[12/13]  
for.end: ; preds = %for.cond  
store i32 1, i32\* %i6, align 4  
br label %for.cond7

[3/4]  
for.body9: ; preds = %for.cond7  
%5 = load i32, i32\* %i6, align 4  
%idxprom10 = zext i32 %5 to i64  
%arrayidx11 = getelementptr inbounds [100 x i32], [100 x i32]\* %a, i64 0, i64 %idxprom10  
%6 = load i32, i32\* %arrayidx11, align 4  
%7 = load i32, i32\* %i6, align 4  
%idxprom12 = zext i32 %7 to i64  
%arrayidx13 = getelementptr inbounds [100 x i32], [100 x i32]\* %b, i64 0, i64 %idxprom12  
%8 = load i32, i32\* %arrayidx13, align 4  
%add = add i32 %6, %8  
%9 = load i32, i32\* %i6, align 4  
%idxprom14 = zext i32 %9 to i64  
%arrayidx15 = getelementptr inbounds [100 x i32], [100 x i32]\* %c, i64 0, i64 %idxprom14  
store i32 %add, i32\* %arrayidx15, align 4  
%10 = load i32, i32\* %i6, align 4  
%sub = sub i32 %10, 1  
%idxprom16 = zext i32 %sub to i64  
%arrayidx17 = getelementptr inbounds [100 x i32], [100 x i32]\* %c, i64 0, i64 %idxprom16  
%11 = load i32, i32\* %arrayidx17, align 4  
%12 = load i32, i32\* %i6, align 4  
%idxprom18 = zext i32 %12 to i64  
%arrayidx19 = getelementptr inbounds [100 x i32], [100 x i32]\* %a, i64 0, i64 %idxprom18  
store i32 %11, i32\* %arrayidx19, align 4  
br label %for.inc20

[14/15]  
entry:  
%retval = alloca i32, align 4  
%a = alloca [100 x i32], align 16  
%b = alloca [100 x i32], align 16  
%c = alloca [100 x i32], align 16  
%i = alloca i32, align 4  
%i6 = alloca i32, align 4  
store i32 0, i32\* %retval, align 4  
%call = call i64 @time(i64\* null) #2  
%conv = trunc i64 %call to i32  
call void @srand(i32 %conv) #2  
store i32 0, i32\* %i, align 4  
br label %for.cond

[16/17]  
for.end22: ; preds = %for.cond7  
ret i32 0

[6/7]  
for.inc: ; preds = %for.body  
%3 = load i32, i32\* %i, align 4  
%inc = add i32 %3, 1  
store i32 %inc, i32\* %i, align 4  
br label %for.cond