; preds = %for.body | | | for.cond: %3 = load i32, i32* %i, align 4 %inc = add i32 %3, 1 store i32 %inc, i32* %i, align 4 br label %for.cond

; preds = %for.inc, %entry | | | for.cond7: %0 = load i 32, i 32*% i, align 4%cmp = icmp ult i32 %0, 100 br i1 %cmp, label %for.body, label %for.end

; preds = %for.inc16, %for.end %4 = load i32, i32* %i6, align 4 %cmp8 = icmp ult i32 %4, 100 br i1 %cmp8, label %for.body9, label %for.end18

```
for.body9:
                                                  ; preds = %for.cond7
                             %5 = load i32, i32* %i6, align 4
%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
                        %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]* %a, i64 0, i64 %idxprom14
                       store i32 %add, i32* %arrayidx15, align 4
                                  br label %for.inc16
```

```
store i32 0, i32* %i, align 4
    br label %for.cond
```

```
%retval = alloca i32, align 4
 %a = 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
```

```
%b = alloca [100 x i32], align 16 | | for.inc16:
                                                                         ; preds = \%for.body9
                                                   %10 = load i32, i32* %i6, align 4
                                                       %inc17 = add i32 %10, 1
                                                   store i32 %inc17, i32* %i6, align 4
                                                         br label %for.cond7
```

```
; preds = %for.cond
store i32 0, i32* %i6, align 4
   br label %for.cond7
```

```
; preds = %for.cond
                for.body:
                             %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
                                                                                     for.end18:
                             %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
```

; preds = %for.cond7

ret i32 0