

```

entry:
  %a = alloca i32*, align 8
  %i = alloca i32, align 4
  %j = alloca i32, align 4
  %k = alloca i32, align 4
  call void @llvm.dbg.declare(metadata i32** %a, metadata !16, metadata !17),
  ... !dbg !18
  %call = call noalias i8* @malloc(i64 16) #5, !dbg !19
  %0 = bitcast i8* %call to i32*, !dbg !20
  store i32* %0, i32** %a, align 8, !dbg !18
  call void @llvm.dbg.declare(metadata i32* %i, metadata !21, metadata !17),
  ... !dbg !22
  call void @llvm.dbg.declare(metadata i32* %j, metadata !23, metadata !17),
  ... !dbg !24
  call void @llvm.dbg.declare(metadata i32* %k, metadata !25, metadata !17),
  ... !dbg !26
  store i32 0, i32* %i, align 4, !dbg !27
  br label %for.cond, !dbg !29

```

```

for.cond:
  %1 = load i32, i32* %i, align 4, !dbg !30
  %cmp = icmp slt i32 %1, 4, !dbg !32
  br i1 %cmp, label %for.body, label %for.end, !dbg !33

```

```

for.body:
  %2 = load i32*, i32** %a, align 8, !dbg !34
  %3 = load i32, i32* %i, align 4, !dbg !36
  %idxprom = sext i32 %3 to i64, !dbg !34
  %arrayidx = getelementptr inbounds i32, i32* %2, i64 %idxprom, !dbg !34
  %4 = load i32, i32* %arrayidx, align 4, !dbg !34
  %call1 = call i32 (i8*, ...) @printf(i8* getelementptr inbounds ([4 x i8],
  ... [4 x i8]* @.str.1.2, i32 0, i32 0), i32 %4), !dbg !37
  br label %for.inc, !dbg !38

```

```

for.inc:
  %5 = load i32, i32* %i, align 4, !dbg !39
  %inc = add nsw i32 %5, 1, !dbg !39
  store i32 %inc, i32* %i, align 4, !dbg !39
  br label %for.cond, !dbg !40, !llvm.loop !41

```

```

for.end:
  ret void, !dbg !43

```

CFG for '_Z26dynamicAllocationArrayTestv' function