实验报告

总体实现思路

该语义实验是基于语法实验构造的语法树完成的。首先基于语法分析的程序,在C++程序中构建语法树。 之后分析语法树,在每一步归约动作后,添加相应的汇编代码逻辑。

汇编代码存储

对于每一步生成的汇编代码,我利用一个全局的vector存储汇编代码语句,并在每一步归约动作处理时将需要添加的代码逻辑添加到该vector中。

变量维护

变量存储

对于变量存储,我将源程序中定义的变量存储在栈中,将定义的全局变量定义在数据段,并利用寄存器%r8-%r15存储表达式中的临时变量。对于寄存器的分配,我利用一个哈希表维护空闲的寄存器。

变量信息

对于每一个变量, 我定义如下结构体:

```
struct Variable {
    string name; // 变量名称
    int offset = 0, size, val; // 变量偏移量, 大小, 数值(常量)
    bool isGlobal = false, isConst; // 是否为全局变量, 是否为常量
    vector<int> arr = vector<int>(); // 数组每一维大小
    string addrReg; // 存储偏移量的寄存器
    string trueName; // 该变量对应的真实变量名(临时变量)
};
```

同时,对于临时变量,需要一个<string、string>的哈希表,用于存储变量名与其对应的寄存器,同时有一个全局的变量tmp对每一个临时变量唯一的命名。

符号表

对于符号表,我用一个栈来维护整个符号表。当进入一个Block时,向栈中添加一个符号表,并在离开 Block时将其弹出。每一个符号表,我用一个<string, Variable>的哈希表维护,通过变量名索引变量信息。

函数调用

对于函数调用,我首先将原先函数中调用的寄存器保存在栈中,之后保存%rbp,最后将%rdi与%rsi中传的参数保存到栈中作为函数中的变量调用。对于每一个函数,我开启一个新的哈希表存储空闲的寄存器。在函数返回时,我将保存的寄存器的值返回,并将存储空闲寄存器的哈希表弹出。

Break与Continue的实现

对于break与continue,我用一个栈来记录当前循环所需要跳转的位置,每次处理doWhile块时,将nextNum与backNum存储在栈中,若在循环中调用了break和continue,则直接跳转到该位置。并在每一个处理完doWhile块时将栈弹出。

实验结果

```
.LC0:
    .string "%11d"
.LC1:
    .string "%11d\n"
    .globl main
main:
            %rbp
    pushq
    pushq
            %rbx
    pushq
            %rcx
    pushq
            %r8
    pushq
            %r9
    pushq
            %r10
    pushq
            %r11
    pushq
            %r12
           %r13
    pushq
    pushq
            %r14
            %r15
    pushq
            %rsp, %rbp
    movq
           $32, %rsp
    subq
    subq
           $16, %rsp
            $16, %rsp
    subq
            -8(%rbp), %r8
    movq
            $8, %rcx
    movq
    subq
            $16, %rsp
    neg
            %rcx
            (%rbp, %rcx), %rax
    leaq
            %rcx
    neg
    movq
            %rax, %rsi
            .LCO(%rip), %rdi
    leaq
            __isoc99_scanf@PLT
    call
    addq
            $16, %rsp
    movq
            -24(%rbp), %r8
            $24, %rcx
    movq
    subq
            $16, %rsp
    neg
            %rcx
            (%rbp, %rcx), %rax
    leaq
            %rcx
    neg
            %rax, %rsi
    movq
           .LCO(%rip), %rdi
    leaq
            __isoc99_scanf@PLT
    call
    addq
            $16, %rsp
            -8(%rbp), %r8
    movq
            $8, %rcx
    movq
            -24(%rbp), %r9
    movq
            $24, %rcx
    movq
            %r9, %r8
    addq
    subq
            $16, %rsp
            %r8, -40(%rbp)
    movq
            -40(%rbp), %r8
    movq
    movq
            $40, %rcx
            $16, %rsp
    subq
```

```
movq
        %r8, %rsi
leaq
        .LC1(%rip), %rdi
call
        printf@PLT
addq
        $16, %rsp
        %rbp, %rsp
movq
         %r15
popq
         %r14
popq
         %r13
popq
         %r12
popq
         %r11
popq
         %r10
popq
         %r9
popq
         %r8
popq
        %rcx
popq
        %rbx
popq
        %rbp
popq
ret
```

```
.LC0:
    .string "%11d"
.LC1:
    .string "%11d\n"
N:
    .long
    .comm
            a, 160, 32
    .globl f
f:
    pushq
            %rbp
    pushq
            %rbx
    pushq
            %rcx
    pushq
            %r8
            %r9
    pushq
    pushq
            %r10
    pushq
            %r11
    pushq
            %r12
            %r13
    pushq
    pushq
            %r14
    pushq
            %r15
            %rsp, %rbp
    movq
            $32, %rsp
    subq
    subq
           $16, %rsp
            %rdi, -8(%rbp)
    movq
            -8(%rbp), %r8
    movq
            $8, %rcx
    movq
    movq
            $1, %r9
            $0, %rdx
    movq
            $1, %rcx
    movq
            %r9, %r8
    cmpq
    cmove
            %rcx, %rdx
            %rdx, %r8
    movq
            $0, %rcx
    movq
    cmpq
            %r8, %rcx
    je .L1
```

```
movq
            $1, %r9
   movq
            %r9, %rax
            %rbp, %rsp
   movq
             %r15
   popq
             %r14
   popq
             %r13
   popq
             %r12
   popq
             %r11
   popq
             %r10
   popq
             %r9
   popq
             %r8
   popq
            %rcx
   popq
            %rbx
   popq
            %rbp
   popq
   ret
.L1:
   movq
            $1, %r9
            $16, %rsp
   subq
            %r9, -24(%rbp)
   movq
            -24(%rbp), %r9
   movq
   movq
            $24, %rcx
            -8(%rbp), %r10
   movq
            $8, %rcx
   movq
            -8(%rbp), %r11
   movq
   movq
            $8, %rcx
            $1, %r12
   movq
            %r12, %r11
   subq
   movq
            %r11, %rdi
   call
            %rax, %r10
   imulq
   neg
            %rcx
            %r10, -24(%rbp)
   movq
   neg
            %rcx
            -24(%rbp), %r9
   movq
            $24, %rcx
   movq
            %r9, %rax
   movq
   movq
            %rbp, %rsp
             %r15
   popq
             %r14
   popq
             %r13
   popq
             %r12
   popq
             %r11
   popq
             %r10
   popq
             %r9
   popq
             %r8
   popq
            %rcx
   popq
            %rbx
   popq
            %rbp
   popq
   ret
            %rbp, %rsp
   movq
             %r15
   popq
             %r14
   popq
             %r13
   popq
             %r12
   popq
             %r11
   popq
             %r10
   popq
```

```
%r9
    popq
    popq
             %r8
            %rcx
    popq
            %rbx
    popq
            %rbp
    popq
    ret
    .globl
            main
main:
    pushq
            %rbp
    pushq
            %rbx
    pushq
            %rcx
            %r8
    pushq
    pushq
            %r9
    pushq
            %r10
    pushq
            %r11
    pushq
            %r12
    pushq
            %r13
    pushq
            %r14
            %r15
    pushq
            %rsp, %rbp
    movq
    subq
            $32, %rsp
            $10, %r8
    movq
            %r8, %rdi
    movq
            f
    call
    subq
            $16, %rsp
            %rax, -8(%rbp)
    movq
            -8(%rbp), %r8
    movq
            $8, %rcx
    movq
    movq
            $3628800, %r9
            $0, %rdx
    movq
            $1, %rcx
    movq
            %r9, %r8
    cmpq
    cmove
            %rcx, %rdx
            %rdx, %r8
    movq
            -8(%rbp), %r9
    movq
            $8, %rcx
    movq
    orq
            %r9, %r8
            $0, %rcx
    movq
            %r8, %rcx
    cmpq
    je .L2
            $2333, %r9
    movq
    subq
            $16, %rsp
            %r9, %rsi
    movq
            .LC1(%rip), %rdi
    leaq
            printf@PLT
    call
            $16, %rsp
    addq
            -8(%rbp), %r9
    movq
            $8, %rcx
    movq
            $0, %rax
    movq
            $1, %rcx
    movq
            %r9, %rax
    cmpq
            %rcx, %rax
    cmove
            %rax, %r9
    movq
            $0, %rcx
    movq
    cmpq
            %r9, %rcx
    je .L4
```

```
-8(%rbp), %r10
   movq
   movq
            $8, %rcx
           -8(%rbp), %r11
   movq
            $8, %rcx
   movq
            $10, %r12
   movq
           %r12, %r11
   subq
           %rcx
   neg
           %r11, -8(%rbp)
   movq
   neg
            %rcx
   jmp .L5
.L4:
           -8(%rbp), %r10
   movq
   movq
           $8, %rcx
           -8(%rbp), %r11
   movq
            $8, %rcx
   movq
   movq
           $10, %r12
   addq
           %r12, %r11
   neg
            %rcx
           %r11, -8(%rbp)
   movq
            %rcx
   neg
.L5:
   jmp .L3
.L2:
   movq
            -8(%rbp), %r10
   movq
            $8, %rcx
           $16, %rsp
   subq
           %r10, %rsi
   movq
           .LC1(%rip), %rdi
   leaq
   call
            printf@PLT
   addq
            $16, %rsp
.L3:
            -8(%rbp), %r10
   movq
   movq
            $8, %rcx
   subq
           $16, %rsp
   movq
           %r10, %rsi
           .LC1(%rip), %rdi
   leaq
   call
            printf@PLT
   addq
            $16, %rsp
            $0, %r10
   movq
            $16, %rsp
   subq
           %r10, -24(%rbp)
   movq
            $0, %r10
   movq
            $16, %rsp
   subq
           %r10, -40(%rbp)
   movq
.L6:
           -24(%rbp), %r10
   movq
            $24, %rcx
   movq
           $3, %r11
   movq
            $0, %rcx
   movq
            $0, %rdx
   movq
            $1, %rcx
   movq
           %r11, %r10
   cmpq
   cmovle %rcx, %rdx
            %rdx, %r10
   movq
   movq
            $0, %rcx
            %r10, %rcx
   cmpq
```

```
je .L7
   movq
           -24(%rbp), %r10
           $24, %rcx
   movq
           $1, %r11
   movq
           $0, %rdx
   movq
           $1, %rcx
   movq
           %r11, %r10
   cmpq
          %rcx, %rdx
   cmove
   movq
           %rdx, %r10
           $0, %rcx
   movq
           %r10, %rcx
   cmpq
   je .L8
   movq
           -24(%rbp), %r11
           $24, %rcx
   movq
           -24(%rbp), %r12
   movq
           $24, %rcx
   movq
           $1, %r13
   movq
           %r13, %r12
   addq
           %rcx
   neg
           %r12, -24(%rbp)
   movq
   neg
           %rcx
   jmp .L6
.L8:
   movq
           -24(%rbp), %r11
   movq
           $24, %rcx
         $3, %r12
   movq
           $0, %rcx
   movq
         $0, %rdx
   movq
   movq
           $1, %rcx
           %r12, %r11
   cmpq
           %rcx, %rdx
   cmove
           %rdx, %r11
   movq
   movq
           $0, %rcx
           %r11, %rcx
   cmpq
   je .L9
   jmp .L7
.L9:
   movq
           -24(%rbp), %r12
           $24, %rcx
   movq
         $16, %rsp
   subq
         %r12, %rsi
   movq
   leaq
           .LC1(%rip), %rdi
   call
           printf@PLT
   addq
           $16, %rsp
   movq
           -24(%rbp), %r12
           $24, %rcx
   movq
           -24(%rbp), %r13
   movq
           $24, %rcx
   movq
           $1, %r14
   movq
           %r14, %r13
   addq
           %rcx
   neg
           %r13, -24(%rbp)
   movq
           %rcx
   neg
    jmp .L6
.L7:
           $3, %r12
   movq
```

```
$0, %rcx
movq
movq
        $3, %r12
        $0, %rcx
movq
        $2, %r13
movq
imulq
       %r13, %r12
subq
        $160, %rsp
        -24(%rbp), %r12
movq
        $24, %rcx
movq
movq
        $1, %r13
        %rcx
neg
        %r13, -24(%rbp)
movq
       %rcx
neg
        -40(%rbp), %r12
movq
        $40, %rcx
movq
        -24(%rbp), %r13
movq
       $24, %rcx
movq
        -24(%rbp), %r14
movq
movq
        $24, %rcx
        %r14, %r13
addq
        %rcx
neg
movq
        %r13, -40(%rbp)
neg
        %rcx
        $0, %r12
movq
        $0, %r13
movq
movq
        $6, %rcx
       %rcx, %r13
imulq
addq
       %r13, %r12
        $0, %r13
movq
movq
        $1, %rcx
       %rcx, %r13
imulq
addq
        %r13, %r12
        $8 , %rcx
movq
imulq
       %rcx, %r12
        $56, %r12
addq
        %r12
neg
movq
        (%rbp, %r12), %r13
neg
        %r12
        -24(%rbp), %r14
movq
        $24, %rcx
movq
       -40(%rbp), %r15
movq
movq
        $40, %rcx
        $2, %rbx
movq
        %rbx, %r15
imulq
        %r15, %r14
addq
neg
        %r12
        %r14, (%rbp, %r12)
movq
        %r12
neg
        $0, %r12
movq
movq
        -24(%rbp), %r13
        $24, %rcx
movq
movq
        $6, %rcx
       %rcx, %r13
imulq
addq
        %r13, %r12
movq
        -40(%rbp), %r13
        $40, %rcx
movq
        $1, %rcx
movq
```

```
imulq
      %rcx, %r13
addq
        %r13, %r12
movq
        $8 , %rcx
       %rcx, %r12
imulq
        $56, %r12
addq
neg
        %r12
        (%rbp, %r12), %r13
movq
        %r12
neg
movq
        $3, %r14
        %r12
neg
       %r14, (%rbp, %r12)
movq
       %r12
neg
        $0, %r12
movq
        $0, %r13
movq
        $6, %rcx
movq
imulq
      %rcx, %r13
addq
       %r13, %r12
        $0, %r13
movq
        $1, %rcx
movq
      %rcx, %r13
imulq
addq
       %r13, %r12
        $8 , %rcx
movq
imulq
        %rcx, %r12
        $56, %r12
addq
neg
        %r12
        (%rbp, %r12), %r13
movq
        %r12
neg
        $16, %rsp
subq
movq
       %r13, %rsi
leaq
       .LC1(%rip), %rdi
call
        printf@PLT
        $16, %rsp
addq
movq
        $0, %r12
       -24(%rbp), %r13
movq
        $24, %rcx
movq
        $6, %rcx
movq
imulq
       %rcx, %r13
addq
        %r13, %r12
        -40(%rbp), %r13
movq
        $40, %rcx
movq
        $1, %rcx
movq
        %rcx, %r13
imulq
addq
        %r13, %r12
        $8 , %rcx
movq
imulq
        %rcx, %r12
        $56, %r12
addq
        %r12
neg
        (%rbp, %r12), %r13
movq
neg
        %r12
        $16, %rsp
subq
        %r13, %rsi
movq
leaq
        .LC1(%rip), %rdi
call
        printf@PLT
addq
        $16, %rsp
movq
        $0, %r12
        -24(%rbp), %r13
movq
```

```
$24, %rcx
movq
movq
        $6, %rcx
imulq
        %rcx, %r13
        %r13, %r12
addq
        -40(%rbp), %r13
movq
movq
        $40, %rcx
        $1, %rcx
movq
       %rcx, %r13
imulq
addq
        %r13, %r12
        $8 , %rcx
movq
imulq
        %rcx, %r12
        $56, %r12
addq
        %r12
neg
        (%rbp, %r12), %r13
movq
        %r12
neg
        $0, %r14
movq
        $0, %r15
movq
        $6, %rcx
movq
        %rcx, %r15
imulq
        %r15, %r14
addq
movq
        $0, %r15
        $1, %rcx
movq
imulq
        %rcx, %r15
addq
        %r15, %r14
movq
        $8 , %rcx
        %rcx, %r14
imulq
addq
        $56, %r14
        %r14
neg
movq
        (%rbp, %r14), %r15
neg
        %r14
addq
        %r15, %r13
        $16, %rsp
subq
movq
       %r13, -216(%rbp)
        -216(%rbp), %r12
movq
        $216, %rcx
movq
        $16, %rsp
subq
movq
       %r12, %rsi
leaq
        .LC1(%rip), %rdi
call
        printf@PLT
        $16, %rsp
addq
        $0, %r12
movq
        $0, %r13
movq
        $6, %rcx
movq
       %rcx, %r13
imulq
addq
        %r13, %r12
        $0, %r13
movq
        $1, %rcx
movq
imulq
       %rcx, %r13
addq
        %r13, %r12
        a(%rip), %rax
leaq
        $8 , %rcx
movq
imulq
        %rcx, %r12
addq
        $0, %r12
        %r12, %rax
addq
movq
        (%rax), %r13
        $5, %r15
movq
```

```
leaq
       a(%rip), %rax
addq
        %r12, %rax
        %r15, (%rax)
movq
        $0, %r12
movq
        $1, %r13
movq
        $6, %rcx
movq
        %rcx, %r13
imulq
        %r13, %r12
addq
movq
        $1, %r13
        $1, %rcx
movq
        %rcx, %r13
imulq
        %r13, %r12
addq
leaq
        a(%rip), %rax
        $8 , %rcx
movq
        %rcx, %r12
imulq
addq
        $0, %r12
addq
        %r12, %rax
        (%rax), %r13
movq
        $2, %r15
movq
leaq
        a(%rip), %rax
addq
        %r12, %rax
        %r15, (%rax)
movq
        $0, %r12
movq
        $0, %r13
movq
movq
        $6, %rcx
        %rcx, %r13
imulq
addq
        %r13, %r12
        $0, %r13
movq
movq
        $1, %rcx
imulq
        %rcx, %r13
addq
        %r13, %r12
        a(%rip), %rax
leaq
movq
        $8 , %rcx
        %rcx, %r12
imulq
addq
        $0, %r12
        %r12, %rax
addq
movq
        (%rax), %r13
        $16, %rsp
subq
        %r13, %rsi
movq
leaq
        .LC1(%rip), %rdi
call
        printf@PLT
        $16, %rsp
addq
        $0, %r12
movq
        $1, %r13
movq
movq
        $6, %rcx
        %rcx, %r13
imulq
        %r13, %r12
addq
movq
        $1, %r13
movq
        $1, %rcx
        %rcx, %r13
imulq
addq
        %r13, %r12
leaq
        a(%rip), %rax
        $8 , %rcx
movq
        %rcx, %r12
imulq
addq
        $0, %r12
        %r12, %rax
addq
```

```
(%rax), %r13
movq
subq
        $16, %rsp
        %r13, %rsi
movq
        .LC1(%rip), %rdi
leaq
        printf@PLT
call
addq
        $16, %rsp
        $0, %r12
movq
        $1, %r13
movq
movq
        $6, %rcx
        %rcx, %r13
imulq
        %r13, %r12
addq
        $1, %r13
movq
movq
        $1, %rcx
        %rcx, %r13
imulq
addq
        %r13, %r12
leaq
        a(%rip), %rax
movq
        $8 , %rcx
        %rcx, %r12
imulq
        $0, %r12
addq
        %r12, %rax
addq
movq
        (%rax), %r13
        $16, %rsp
subq
leaq
        a(%rip), %rax
addq
        %r12, %rax
movq
       %rax, %rsi
        .LCO(%rip), %rdi
leaq
        __isoc99_scanf@PLT
call
        $16, %rsp
addq
movq
        $0, %r12
        $1, %r13
movq
        $6, %rcx
movq
        %rcx, %r13
imulq
addq
        %r13, %r12
        $1, %r13
movq
        $1, %rcx
movq
       %rcx, %r13
imulq
addq
        %r13, %r12
leaq
        a(%rip), %rax
        $8 , %rcx
movq
       %rcx, %r12
imulq
        $0, %r12
addq
addq
        %r12, %rax
        (%rax), %r13
movq
        $16, %rsp
subq
movq
        %r13, %rsi
        .LC1(%rip), %rdi
leaq
call
        printf@PLT
addq
        $16, %rsp
movq
        $0, %r12
        %r12, %rax
movq
        %rbp, %rsp
movq
         %r15
popq
         %r14
popq
         %r13
popq
         %r12
popq
         %r11
popq
```

```
%r10
popq
popq
         %r9
         %r8
popq
        %rcx
popq
        %rbx
popq
        %rbp
popq
ret
        %rbp, %rsp
movq
popq
         %r15
         %r14
popq
         %r13
popq
         %r12
popq
         %r11
popq
         %r10
popq
         %r9
popq
         %r8
popq
        %rcx
popq
        %rbx
popq
        %rbp
popq
ret
```

```
.LC0:
    .string "%11d"
.LC1:
    .string "%11d\n"
N:
    .long
            10
            a,96,32
    .comm
    .comm
            b,96,32
            c,96,32
    .comm
    .globl
            main
main:
    pushq
            %rbp
    pushq
            %rbx
    pushq
            %rcx
    pushq
            %r8
    pushq
            %r9
    pushq
            %r10
    pushq
            %r11
            %r12
    pushq
    pushq
            %r13
            %r14
    pushq
            %r15
    pushq
            %rsp, %rbp
    movq
    subq
            $32, %rsp
    subq
            $16, %rsp
            $16, %rsp
    subq
            -8(%rbp), %r8
    movq
            $8, %rcx
    movq
            $16, %rsp
    subq
    neg
    leaq
            (%rbp, %rcx), %rax
    neg
            %rcx
```

```
%rax, %rsi
   movq
   leaq
           .LCO(%rip), %rdi
   call
           __isoc99_scanf@PLT
           $16, %rsp
   addq
           -24(%rbp), %r8
   movq
   movq
           $24, %rcx
           $16, %rsp
   subq
           %rcx
   neg
           (%rbp, %rcx), %rax
   leaq
           %rcx
   neg
   movq
           %rax, %rsi
   leaq
           .LC0(%rip), %rdi
   call
            __isoc99_scanf@PLT
           $16, %rsp
   addq
           $0, %r8
   movq
           $16, %rsp
   subq
           %r8, -40(%rbp)
   movq
.L1:
           -40(%rbp), %r8
   movq
           $40, %rcx
   movq
   movq
           -8(%rbp), %r9
           $8, %rcx
   movq
           $0, %rdx
   movq
           $1, %rcx
   movq
   cmpq
           %r9, %r8
           %rcx, %rdx
   cmovl
           %rdx, %r8
   movq
           $0, %rcx
   movq
   cmpq
           %r8, %rcx
   je .L2
           $0, %r8
   movq
           -40(%rbp), %r9
   movq
   movq
           $40, %rcx
           $1, %rcx
   movq
           %rcx, %r9
   imulq
           %r9, %r8
   addq
   leaq
           a(%rip), %rax
   movq
           $8 , %rcx
           %rcx, %r8
   imulq
           $0, %r8
   addq
   addq
           %r8, %rax
           (%rax), %r9
   movq
           $16, %rsp
   subq
   leaq
           a(%rip), %rax
   addq
           %r8, %rax
   movq
           %rax, %rsi
   leaq
           .LC0(%rip), %rdi
   call
            __isoc99_scanf@PLT
   addq
           $16, %rsp
   movq
           -40(%rbp), %r8
           $40, %rcx
   movq
           -40(%rbp), %r9
   movq
           $40, %rcx
   movq
            $1, %r10
   movq
           %r10, %r9
   addq
           %rcx
   neg
```

```
movq %r9, -40(%rbp)
   neg
           %rcx
   jmp .L1
.L2:
           -40(%rbp), %r8
   movq
   movq
           $40, %rcx
           $0, %r9
   movq
           %rcx
   neg
   movq
           %r9, -40(%rbp)
           %rcx
   neg
.L3:
           -40(%rbp), %r8
   movq
   movq
           $40, %rcx
           -24(%rbp), %r9
   movq
           $24, %rcx
   movq
          $0, %rdx
   movq
           $1, %rcx
   movq
           %r9, %r8
   cmpq
           %rcx, %rdx
   cmovl
           %rdx, %r8
   movq
   movq
           $0, %rcx
           %r8, %rcx
   cmpq
   je .L4
   movq
           $0, %r8
   movq
           -40(%rbp), %r9
          $40, %rcx
   movq
   movq
           $1, %rcx
   imulq %rcx, %r9
   addq
           %r9, %r8
           b(%rip), %rax
   leaq
   movq
           $8 , %rcx
   imulq %rcx, %r8
   addq
           $0, %r8
   addq
           %r8, %rax
   movq
           (%rax), %r9
           $16, %rsp
   subq
   leaq
           b(%rip), %rax
   addq
           %r8, %rax
           %rax, %rsi
   movq
           .LCO(%rip), %rdi
   leaq
   call
            __isoc99_scanf@PLT
           $16, %rsp
   addq
           -40(%rbp), %r8
   movq
          $40, %rcx
   movq
   movq
           -40(%rbp), %r9
           $40, %rcx
   movq
           $1, %r10
   movq
   addq
           %r10, %r9
   neg
           %rcx
           %r9, -40(%rbp)
   movq
   neg
           %rcx
   jmp .L3
.L4:
            $0, %r8
   movq
           $16, %rsp
   subq
           %r8, -56(%rbp)
   movq
```

```
-40(%rbp), %r8
   movq
   movq
           $40, %rcx
           $0, %r9
   movq
           %rcx
   neg
           %r9, -40(%rbp)
   movq
   neg
           %rcx
.L5:
           -40(%rbp), %r8
   movq
   movq
           $40, %rcx
           -8(%rbp), %r9
   movq
           $8, %rcx
   movq
          $0, %rdx
   movq
   movq
           $1, %rcx
           %r9, %r8
   cmpq
           %rcx, %rdx
   cmovl
           %rdx, %r8
   movq
           $0, %rcx
   movq
           %r8, %rcx
   cmpq
   je .L6
   movq
           -56(%rbp), %r8
   movq
           $56, %rcx
           $0, %r9
   movq
   neg
           %rcx
           %r9, -56(%rbp)
   movq
   neg
           %rcx
.L7:
           -56(%rbp), %r8
   movq
           $56, %rcx
   movq
   movq
           -24(%rbp), %r9
           $24, %rcx
   movq
           $0, %rdx
   movq
           $1, %rcx
   movq
   cmpq
           %r9, %r8
           %rcx, %rdx
   cmovl
           %rdx, %r8
   movq
           $0, %rcx
   movq
   cmpq
           %r8, %rcx
   je .L8
           $0, %r8
   movq
           -40(%rbp), %r9
   movq
           $40, %rcx
   movq
           $1, %rcx
   movq
           %rcx, %r9
   imulq
           %r9, %r8
   addq
   leaq
           a(%rip), %rax
           $8 , %rcx
   movq
           %rcx, %r8
   imulq
   addq
           $0, %r8
   addq
           %r8, %rax
           (%rax), %r9
   movq
           $0, %r10
   movq
           -56(%rbp), %r11
   movq
           $56, %rcx
   movq
           $1, %rcx
   movq
           %rcx, %r11
   imulq
           %r11, %r10
   addq
```

```
leaq
           b(%rip), %rax
   movq
           $8 , %rcx
   imulq
           %rcx, %r10
   addq
           $0, %r10
           %r10, %rax
   addq
   movq
           (%rax), %r11
           %r11, %r9
   imulq
           $16, %rsp
   subq
           %r9, -72(%rbp)
   movq
           $0, %r8
   movq
           -40(%rbp), %r9
   movq
           $40, %rcx
   movq
           -56(%rbp), %r11
   movq
   movq
           $56, %rcx
           %r11, %r9
   addq
           $1, %rcx
   movq
   imulq
           %rcx, %r9
           %r9, %r8
   addq
   leaq
           c(%rip), %rax
           $8 , %rcx
   movq
   imulq
           %rcx, %r8
           $0, %r8
   addq
   addq
           %r8, %rax
           (%rax), %r9
   movq
   movq
           $0, %r11
   movq
           -40(%rbp), %r12
           $40, %rcx
   movq
           -56(%rbp), %r13
   movq
   movq
           $56, %rcx
           %r13, %r12
   addq
           $1, %rcx
   movq
          %rcx, %r12
   imulq
   addq
           %r12, %r11
   leaq
           c(%rip), %rax
           $8 , %rcx
   movq
          %rcx, %r11
   imulq
   addq
           $0, %r11
   addq
           %r11, %rax
           (%rax), %r12
   movq
           -72(%rbp), %r13
   movq
   movq
           $72, %rcx
           %r13, %r12
   addq
   leaq
           c(%rip), %rax
   addq
           %r8, %rax
   movq
           %r12, (%rax)
           -56(%rbp), %r8
   movq
           $56, %rcx
   movq
           -56(%rbp), %r9
   movq
   movq
           $56, %rcx
           $1, %r11
   movq
   addq
           %r11, %r9
           %rcx
   neg
   movq
           %r9, -56(%rbp)
   neg
           %rcx
   jmp .L7
.L8:
```

```
-40(%rbp), %r8
   movq
   movq
           $40, %rcx
           -40(%rbp), %r9
   movq
           $40, %rcx
   movq
           $1, %r11
   movq
   addq
           %r11, %r9
           %rcx
   neg
           %r9, -40(%rbp)
   movq
           %rcx
   neg
   jmp .L5
.L6:
           -40(%rbp), %r8
   movq
   movq
           $40, %rcx
           $0, %r9
   movq
           %rcx
   neg
           %r9, -40(%rbp)
   movq
           %rcx
   neg
.L9:
           -40(%rbp), %r8
   movq
           $40, %rcx
   movq
   movq
           -8(%rbp), %r9
           $8, %rcx
   movq
   movq
           -24(%rbp), %r11
           $24, %rcx
   movq
   addq
           %r11, %r9
           $1, %r11
   movq
           %r11, %r9
   subq
           $0, %rdx
   movq
   movq
           $1, %rcx
           %r9, %r8
   cmpq
           %rcx, %rdx
   cmovl
           %rdx, %r8
   movq
   movq
           $0, %rcx
           %r8, %rcx
   cmpq
   je .L10
           $0, %r8
   movq
   movq
           -40(%rbp), %r9
           $40, %rcx
   movq
           $1, %rcx
   movq
          %rcx, %r9
   imulq
           %r9, %r8
   addq
   leaq
           c(%rip), %rax
           $8 , %rcx
   movq
          %rcx, %r8
   imulq
           $0, %r8
   addq
           %r8, %rax
   addq
           (%rax), %r9
   movq
   subq
           $16, %rsp
   movq
           %r9, %rsi
           .LC1(%rip), %rdi
   leaq
           printf@PLT
   call
   addq
           $16, %rsp
           -40(%rbp), %r8
   movq
            $40, %rcx
   movq
           -40(%rbp), %r9
   movq
           $40, %rcx
   movq
```

```
movq
         $1, %r11
   addq
            %r11, %r9
   neg
            %rcx
            %r9, -40(%rbp)
   movq
            %rcx
   neg
   jmp .L9
.L10:
   movq
            $0, %r8
   movq
            %r8, %rax
   movq
            %rbp, %rsp
            %r15
   popq
            %r14
   popq
   popq
             %r13
             %r12
   popq
             %r11
   popq
             %r10
   popq
             %r9
   popq
            %r8
   popq
            %rcx
   popq
            %rbx
   popq
   popq
            %rbp
   ret
            %rbp, %rsp
   movq
            %r15
   popq
   popq
             %r14
             %r13
   popq
             %r12
   popq
             %r11
   popq
   popq
             %r10
             %r9
   popq
             %r8
   popq
            %rcx
   popq
   popq
            %rbx
            %rbp
   popq
   ret
```

```
.LC0:
    .string "%11d"
.LC1:
    .string "%11d\n"
    .globl main
main:
    pushq
            %rbp
    pushq
            %rbx
    pushq
            %rcx
    pushq
            %r8
    pushq
            %r9
            %r10
    pushq
    pushq
            %r11
    pushq
            %r12
            %r13
    pushq
    pushq
            %r14
    pushq
            %r15
```

```
%rsp, %rbp
   movq
   subq
            $32, %rsp
            $6, %r8
   movq
            $6, %r8
   movq
            $304, %rsp
   subq
           $1, %r8
   movq
            $16, %rsp
   subq
           %r8, -312(%rbp)
   movq
            $1, %r8
   movq
            $16, %rsp
   subq
           %r8, -328(%rbp)
   movq
.L1:
   movq
           -312(%rbp), %r8
           $312, %rcx
   movq
           $5, %r9
   movq
           $0, %rdx
   movq
            $1, %rcx
   movq
           %r9, %r8
   cmpq
           %rcx, %rdx
   cmov1
           %rdx, %r8
   movq
   movq
           -328(%rbp), %r9
           $328, %rcx
   movq
           $5, %r10
   movq
           $0, %rdx
   movq
   movq
           $1, %rcx
           %r10, %r9
   cmpq
           %rcx, %rdx
   cmov1
           %rdx, %r9
   movq
   andq
           %r9, %r8
            $0, %rcx
   movq
           %r8, %rcx
   cmpq
   je .L2
   movq
            $0, %r8
           -312(%rbp), %r9
   movq
            $312, %rcx
   movq
           $6, %rcx
   movq
   imulq
           %rcx, %r9
           %r9, %r8
   addq
           -328(%rbp), %r9
   movq
           $328, %rcx
   movq
            $1, %rcx
   movq
           %rcx, %r9
   imulq
   addq
           %r9, %r8
            $8 , %rcx
   movq
   imulq
           %rcx, %r8
            $8, %r8
   addq
   neg
            %r8
           (%rbp, %r8), %r9
   movq
   neg
           %r8
           -312(%rbp), %r10
   movq
   movq
            $312, %rcx
           -328(%rbp), %r11
   movq
            $328, %rcx
   movq
            %r11, %r10
   addq
            %r8
   neg
            %r10, (%rbp, %r8)
   movq
```

```
%r8
   neg
   movq
           -328(%rbp), %r8
           $328, %rcx
   movq
           -328(%rbp), %r9
   movq
   movq
           $328, %rcx
           $1, %r10
   movq
   addq
           %r10, %r9
           %rcx
   neg
           %r9, -328(%rbp)
   movq
           %rcx
   neg
   jmp .L1
.L2:
.L3:
           -312(%rbp), %r8
   movq
   movq
           $312, %rcx
         $5, %r9
   movq
           $0, %rdx
   movq
           $1, %rcx
   movq
           %r9, %r8
   cmpq
           %rcx, %rdx
   cmovl
   movq
           %rdx, %r8
           -328(%rbp), %r9
   movq
   movq
           $328, %rcx
          $6, %r10
   movq
   movq
           $0, %rdx
           $1, %rcx
   movq
           %r10, %r9
   cmpq
           %rcx, %rdx
   cmovl
           %rdx, %r9
   movq
           %r9, %r8
   andq
           $0, %rcx
   movq
           %r8, %rcx
   cmpq
   je .L4
           $0, %r8
   movq
           -312(%rbp), %r9
   movq
           $312, %rcx
   movq
   movq
           $6, %rcx
   imulq
          %rcx, %r9
   addq
           %r9, %r8
           -328(%rbp), %r9
   movq
   movq
           $328, %rcx
           $1, %rcx
   movq
           %rcx, %r9
   imulq
           %r9, %r8
   addq
   movq
           $8 , %rcx
   imulq
           %rcx, %r8
   addq
           $8, %r8
           %r8
   neg
   movq
           (%rbp, %r8), %r9
   neg
           %r8
           -312(%rbp), %r10
   movq
           $312, %rcx
   movq
           -328(%rbp), %r11
   movq
           $328, %rcx
   movq
   subq
           %r11, %r10
           %r8
   neg
```

```
%r10, (%rbp, %r8)
   movq
   neg
            %r8
            -312(%rbp), %r8
   movq
           $312, %rcx
   movq
           -312(%rbp), %r9
   movq
           $312, %rcx
   movq
            $1, %r10
   movq
           %r10, %r9
   addq
            %rcx
   neg
           %r9, -312(%rbp)
   movq
   neg
            %rcx
   jmp .L3
.L4:
            $0, %r8
   movq
            $1, %r9
   movq
   movq
            $6, %rcx
   imulq
           %rcx, %r9
           %r9, %r8
   addq
            $1, %r9
   movq
            $1, %rcx
   movq
   imulq
           %rcx, %r9
           %r9, %r8
   addq
   movq
            $8 , %rcx
           %rcx, %r8
   imulq
   addq
            $8, %r8
   neg
            %r8
            (%rbp, %r8), %r9
   movq
           %r8
   neg
   movq
            $0, %r10
            $1, %r11
   movq
            $6, %rcx
   movq
           %rcx, %r11
   imulq
   addq
           %r11, %r10
           $4, %r11
   movq
            $1, %rcx
   movq
          %rcx, %r11
   imulq
   addq
           %r11, %r10
            $8 , %rcx
   movq
           %rcx, %r10
   imulq
            $8, %r10
   addq
   neg
           %r10
            (%rbp, %r10), %r11
   movq
            %r10
   neg
           %r11, %r9
   addq
            $0, %r8
   movq
            $4, %r11
   movq
            $6, %rcx
   movq
   imulq
           %rcx, %r11
   addq
           %r11, %r8
            $5, %r11
   movq
            $1, %rcx
   movq
   imulq
           %rcx, %r11
   addq
           %r11, %r8
            $8 , %rcx
   movq
   imulq
            %rcx, %r8
   addq
            $8, %r8
```

```
neg
        %r8
movq
        (%rbp, %r8), %r11
        %r8
neg
       %r11, %r9
addq
subq
        $16, %rsp
movq
       %r9, %rsi
       .LC1(%rip), %rdi
leaq
       printf@PLT
call
addq
        $16, %rsp
       %rbp, %rsp
movq
        %r15
popq
        %r14
popq
popq
        %r13
        %r12
popq
        %r11
popq
popq
        %r10
popq
        %r9
        %r8
popq
       %rcx
popq
        %rbx
popq
       %rbp
popq
ret
```