Mode: All

Left file: D:\workspace\study\Code-Optimization\MIT-6.172\Lectures\Week3\MIT6_172F18_hw3\homework3\loop-nv-nu.s Right file: D:\workspace\study\Code-Optimization\MIT-6.172\Lectures\Week3\MIT6_172F18_hw3\homework3\loop-v-nu.s

Right file: D:\workspace\study\Code-Optimization\MIT	-6.172\L€	ectur	res\Week3\MIT6_172F18_hw3\homework3\loop-v-nu.s		
.text		=	.text		
.file "loop.c"			.file "loop.c"		
.section .rodata.cst8,"a	aM",@р		.section .rodata.cst8,"a	0, "٩	др
» rogbits,8			» rogbits,8		
.p2align 3	#		.p2align 3	# -	
<pre>» Begin function main</pre>			<pre>» Begin function main</pre>		
.LCPIO_0:			.LCPI0_0:		
.quad 4472406533629990549	# do		.quad 4472406533629990549	# 0	40
·	# uo			# (10
» uble 1.00000000000001E-9			» uble 1.000000000000001E-9		
.text			.text		
.globl main			.globl main		
.p2align 4, 0x90			.p2align 4, 0x90		
.type main,@function			.type main,@function		
main:	# @m		main:	# @	am 🖟
» ain			» ain		
.cfi_startproc			.cfi_startproc		
# %bb.0:			# %bb.0:		
···			pushq %r15		
.cfi_def_cfa_offset 16			.cfi_def_cfa_offset 16		
pushq %r14			pushq %r14		
.cfi_def_cfa_offset 24			.cfi_def_cfa_offset 24		
pushq %r12			pushq %r12		
.cfi_def_cfa_offset 32			.cfi_def_cfa_offset 32		
pushq %rbx			pushq %rbx		
.cfi_def_cfa_offset 40			.cfi_def_cfa_offset 40		
subq \$12328, %rsp	# im		subq \$12328, %rsp	# j	im
» m = 0x3028			» m = 0x3028		
.cfi_def_cfa_offset 12368			.cfi_def_cfa_offset 12368		
.cfi_offset %rbx, -40			.cfi_offset %rbx, -40		
			.cfi_offset %r12, -32		
.cfi_offset %r12, -32			_		
.cfi_offset %r14, -24			.cfi_offset %r14, -24		
.cfi_offset %r15, -16			.cfi_offset %r15, -16		
movl \$0, 12(%rsp)			movl \$0, 12(%rsp)		
leaq 8224(%rsp), %rdi			leaq 8224(%rsp), %rdi		
xorl %ebx, %ebx			xorl %ebx, %ebx		
xorl %esi, %esi			xorl %esi, %esi		
movl \$4096, %edx	# im		movl \$4096, %edx	# j	im
» m = 0x1000			» m = 0x1000		
callq memset			callq memset		
leaq 4128(%rsp), %rdi			leaq 4128(%rsp), %rdi		
xorl %esi, %esi			xorl %esi, %esi		
	#			щ.	
movl \$4096, %edx	# im		movl \$4096, %edx	# i	LM
» m = 0x1000			» m = 0x1000		
callq memset			callq memset		
leaq 32(%rsp), %rdi			leaq 32(%rsp), %rdi		
xorl %esi, %esi			xorl %esi, %esi		
movl \$4096, %edx	# im		movl \$4096, %edx	# j	im
$m = 0 \times 1000$			$m = 0 \times 1000$		
callq memset			callq memset		
leaq 16(%rsp), %rsi			leaq 16(%rsp), %rsi		
movl \$1, %edi			movl \$1, %edi		
callq clock_gettime			callq clock_gettime		
movq 16(%rsp), %r15			movq 16(%rsp), %r15		
movq 24(%rsp), %r14			movq 24(%rsp), %r14		
.p2align 4, 0x90			.p2align 4, 0x90		
.LBB0_1:	# =>		.LBB0_1:	# =	=>

Beyond Compare v3.3.13

(continued)			1	1		
» This Loop Header	: Depth=1			» This Loop Header: Depth=1		
		#			#	
» Child Loop BB0				<pre>» Child Loop BB0_2 Depth 2</pre>		
movq \$-	4096, %rax	# im		movq \$-4096, %rax	#	im
m = 0xF000				$m = 0 \times F000$		
.p2align	4, 0x90			.p2align 4, 0x90		
.LBB0_2:		#		.LBB0_2:	#	
» Parent Loop BB0_	1 Depth=1			» Parent Loop BBO_1 Depth=1		
	•	# =>			#	=>
» This Inner Loo	p Header: Depth=2	,		<pre>» This Inner Loop Header: Depth=2</pre>		•
	24(%rsp,%rax), %ecx		<>	movdqa 8224(%rsp,%rax), %xmm0		
	320(%rsp,%rax), %ecx			, , , , , , , , , , , , , , , , , , , ,		
	cx, 4128(%rsp,%rax)			movdqa %xmm0, 4128(%rsp,%rax)		
	, %rax			addq \$ <mark>16</mark> , %rax		
1	BB0_2		=	jne .LBB0_2		
# %bb.3:		#		# %bb.3:	#	
<pre>» in Loop: Header=BB0_1 Depth=1</pre>				<pre>» in Loop: Header=BB0_1 Depth=1</pre>		
addl \$1	, %ebx			addl \$1, %ebx		
cmpl \$1	00000, %ebx	# im		cmpl \$100000, %ebx	#	im
» m = 0x186A0	•			» m = 0x186A0		
	BB0_1			jne .LBB0_1		
# %bb.4:				# %bb.4:		
	(%rsp), %rsi			leaq 16(%rsp), %rsi		
1						
	, %edi			movl \$1, %edi		
	ock_gettime			callq clock_gettime		
1	(%rsp), %r12			movq 16(%rsp), %r12		
1	15, %r12			subq %r15, %r12		
movq 24	(%rsp), %rbx			movq 24(%rsp), %rbx		
subq %r	14, %rbx			subq %r14, %rbx		
leaq 12	(%rsp), %rdi			leaq 12(%rsp), %rdi		
callq ra	nd_r			callq rand_r		
movl %e	ax, %ecx			movl %eax, %ecx		
	1, %ecx			sarl \$31, %ecx		
	2, %ecx			shrl \$22, %ecx		
	ax, %ecx			addl %eax, %ecx		
	1024, %ecx	# im		andl \$-1024, %ecx	#	im
» m = 0xFC00	1024, %ECX	π 1111		» m = 0xFC00	π	TIII
	9/					
	cx, %eax			subl %ecx, %eax		
cltq				cltq		
cvtsi2sdq	%r12, %xmm1			cvtsi2sdq %r12, %xmm1		
			-+	xorps %xmm0, %xmm0		
cvtsi2sdq	%rbx, %xmm0		=	cvtsi2sdq %rbx, %xmm0		
	(%rsp,%rax,4), %ebx			movl 32(%rsp,%rax,4), %ebx		
mulsd .L	CPI0_0(%rip), %xmm0			mulsd .LCPIO_0(%rip), %xmm0		
addsd %x	mm1, %xmm0			addsd %xmm1, %xmm0		
movl \$.	L.str, %edi			movl \$.L.str, %edi		
	024, %esi	# im		movl \$1024, %esi	#	im
» m = 0x400	-			» m = 0x400		
	00000, %edx	# im		movl \$100000, %edx	#	im
» m = 0x186A0	,	1111		» m = 0x186A0		
	I str 1 %acv					
•	L.str.1, %ecx			movl \$.L.str.1, %ecx		
	L.str.2, %r8d			mov1 \$.L.str.2, %r8d		
	, %al			movb \$1, %al		
	intf			callq printf		
	bx, %eax			movl %ebx, %eax		
addq \$1	2328, %rsp	# im		addq \$12328, %rsp		im
				Beyond Comp	are v	/3.3.13

```
m = 0x3028
                                                   m = 0x3028
        popq
                %rbx
                                                                   %rbx
                                                           popq
                %r12
                                                                   %r12
        popq
                                                           popq
                %r14
                                                                   %r14
        popq
                                                           popq
                %r15
                                                                   %r15
        popq
                                                           popq
        retq
                                                           retq
.Lfunc_end0:
                                                   .Lfunc_end0:
                main, .Lfunc_end0-main
                                                                   main, .Lfunc_end0-main
        .size
                                                           .size
        .cfi_endproc
                                                           .cfi_endproc
                                         # --
                                                                                             # --
   End function
                                                     End function
        .type
                .L.str,@object
                                         # @.
                                                           .type
                                                                   .L.str,@object
                                                                                             # @.
» str
                                                   » str
                         .rodata.str1.1, "aMS"
                                                                            .rodata.str1.1, "aMS"
        .section
                                                           .section
» ,@progbits,1
                                                   » ,@progbits,1
.L.str:
                                                   .L.str:
        .asciz "Elapsed execution time: %f
                                                           .asciz "Elapsed execution time: %f
» sec; N: %d, I: %d, __OP__: %s, __TYPE__: %
                                                   » sec; N: %d, I: %d, __OP__: %s, __TYPE__: %
» s\n"
                                                   » s\n"
                .L.str, 72
                                                                    .L.str, 72
        .size
                                                           .size
                                                                                             # @.
                .L.str.1,@object
                                         # @.
                                                                    .L.str.1,@object
        .type
                                                           .type
» str.1
                                                   » str.1
.L.str.1:
                                                   .L.str.1:
                                                                   "+"
                "+"
        .asciz
                                                            .asciz
        .size
                .L.str.1, 2
                                                           .size
                                                                   .L.str.1, 2
                .L.str.2,@object
                                                                    .L.str.2,@object
        .type
                                         # @.
                                                           .type
                                                                                             # @.
» str.2
                                                   » str.2
.L.str.2:
                                                   .L.str.2:
                "uint32 t"
                                                                   "uint32_t"
        .asciz
                                                           .asciz
        .size
                .L.str.2, 9
                                                           .size
                                                                   .L.str.2, 9
        .ident "clang version 6.0.0-1ubuntu
                                                           .ident "clang version 6.0.0-1ubuntu
» 2 (tags/RELEASE 600/final)"
                                                   » 2 (tags/RELEASE 600/final)"
                         ".note.GNU-stack",""
                                                                            ".note.GNU-stack",""
        .section
                                                           .section
» ,@progbits
                                                   » ,@progbits
```