中间代码：

const int c1 = 3

const int c2 = 4

const char c3 = 115

const char c4 = 116

var int v1

array int v2[13]

var char v3

array char v4[15]

func int Fib

para int n

n == 0

bz LABEL0

ret 1

LABEL0:

n == 1

bz LABEL1

ret 1

LABEL1:

$VAR0 = n - 1

push $VAR0

$VAR1 = call Fib

$VAR2 = n - 2

push $VAR2

$VAR3 = call Fib

$VAR4 = $VAR1 + $VAR3

ret $VAR4

end func

func void ifst

para int c

c == 4

bz LABEL2

print "This number is equal to4.\n"

LABEL2:

c >= 1

bz LABEL3

print "This number is great than or equal to1.\n"

LABEL3:

$VAR0 = 3 \* 4

c < $VAR0

bz LABEL4

print "This number is smaller than 12.\n"

LABEL4:

$VAR0 = v2[4]

c <= $VAR0

bz LABEL5

print "This number is smaller than or equal to 5.\n"

LABEL5:

push 2

$VAR0 = call Fib

c > $VAR0

bz LABEL6

print "This number is great than 2.\n"

LABEL6:

c != 3

bz LABEL7

print "This number is not equal to c1.\n"

LABEL7:

ret

ret

end func

func void whilest

para int t

para char c

var int s

s = t

LABEL8:

t >= 0

bz LABEL9

print "count:"

$VAR0 = s - t

$VAR1 = $VAR0 + 1

print $VAR1

print "\n"

print c

print "\n"

$VAR0 = t - 1

t = $VAR0

goto LABEL8

LABEL9:

ret

ret

end func

func void switchst

v1 == 10

bz LABEL11

$VAR0 = v1 + 1

v1 = $VAR0

print 115

print "\n"

goto LABEL10

LABEL11:

$VAR0 == 8

bz LABEL12

$VAR0 = v1 - 2

v1 = $VAR0

$VAR0 = v1 + 2

print $VAR0

print "\n"

goto LABEL10

LABEL12:

print "v1 does not equal 10 or equal to 8;\n"

goto LABEL10

LABEL10:

print "v1 is equal to "

print v1

print "\n"

ret

ret

end func

func void main

var int aaa

var int \_a2

var int a3

var char a4

var char a5

var char a6

aaa = 0

LABEL13:

aaa != 13

bz LABEL14

push aaa

$VAR0 = call Fib

v2[aaa] = $VAR0

$VAR0 = aaa + 1

aaa = $VAR0

goto LABEL13

LABEL14:

scan \_a2

push \_a2

call ifst

scan a3

scan a4

push a3

push a4

call whilest

$VAR0 = 6 \* 4

$VAR1 = $VAR0 / 8

$VAR2 = \_a2 + $VAR1

$VAR3 = $VAR2 - 1

$VAR4 = $VAR3 + 2

v1 = $VAR4

$VAR0 = v2[v1]

print $VAR0

print "\n"

call switchst

end

end

汇编代码：

.data

c1: .word 3

c2: .word 4

c3: .word 115

c4: .word 116

v1: .word 0

v2: .space 52

v3: .word 0

v4: .space 60

STR31: .asciiz "This number is equal to4.\n"

STR35: .asciiz "This number is great than or equal to1.\n"

STR40: .asciiz "This number is smaller than 12.\n"

STR45: .asciiz "This number is smaller than or equal to 5.\n"

STR51: .asciiz "This number is great than 2.\n"

STR55: .asciiz "This number is not equal to c1.\n"

STR68: .asciiz "count:"

STR72: .asciiz "\n"

STR74: .asciiz "\n"

STR88: .asciiz "\n"

STR97: .asciiz "\n"

STR100: .asciiz "v1 does not equal 10 or equal to 8;\n"

STR103: .asciiz "v1 is equal to "

STR105: .asciiz "\n"

STR143: .asciiz "\n"

.text

j main

Fib:

li $t0,0

lw $s4,0($sp)

subi $sp,$sp,8

sw $s4,0($sp)

addi $sp,$sp,12

move $s1,$fp

move $fp,$sp

addi $fp,$fp,-4

sw $s1,0($fp)

sw $ra,-4($fp)

addi $sp,$sp,-8

li $t1,6

TEP\_0:

lw $0,0($sp)

addi $sp,$sp,-4

addi $t0,$t0,1

bne $t0,$t1,TEP\_0

lw $t0,-8($fp)

li $t1,0

sub $t2,$t0,$t1

bne $t2,$0,LABEL0

li $v0,1

lw $ra,-4($fp)

addi $sp,$fp,4

lw $fp,0($fp)

jr $ra

LABEL0:

lw $t0,-8($fp)

li $t1,1

sub $t2,$t0,$t1

bne $t2,$0,LABEL1

li $v0,1

lw $ra,-4($fp)

addi $sp,$fp,4

lw $fp,0($fp)

jr $ra

LABEL1:

lw $t0,-8($fp)

li $t1,1

sub $t0,$t0,$t1

sw $t0,-12($fp)

lw $t0,-12($fp)

subi $sp,$sp,4

sw $t0,0($sp)

jal Fib

move $t0,$v0

sw $t0,-16($fp)

lw $t0,-8($fp)

li $t1,2

sub $t0,$t0,$t1

sw $t0,-20($fp)

lw $t0,-20($fp)

subi $sp,$sp,4

sw $t0,0($sp)

jal Fib

move $t0,$v0

sw $t0,-24($fp)

lw $t0,-16($fp)

lw $t1,-24($fp)

add $t0,$t0,$t1

sw $t0,-28($fp)

lw $v0,-28($fp)

lw $ra,-4($fp)

addi $sp,$fp,4

lw $fp,0($fp)

jr $ra

ifst:

li $t0,0

lw $s4,0($sp)

subi $sp,$sp,8

sw $s4,0($sp)

addi $sp,$sp,12

move $s1,$fp

move $fp,$sp

addi $fp,$fp,-4

sw $s1,0($fp)

sw $ra,-4($fp)

addi $sp,$sp,-8

li $t1,2

TEP\_1:

lw $0,0($sp)

addi $sp,$sp,-4

addi $t0,$t0,1

bne $t0,$t1,TEP\_1

lw $t0,-8($fp)

li $t1,4

sub $t2,$t0,$t1

bne $t2,$0,LABEL2

la $a0,STR31

li $v0,4

syscall

LABEL2:

lw $t0,-8($fp)

li $t1,1

sub $t2,$t0,$t1

bltz $t2,LABEL3

la $a0,STR35

li $v0,4

syscall

LABEL3:

li $t0,3

li $t1,4

mult $t0,$t1

mflo $t0

sw $t0,-12($fp)

lw $t0,-8($fp)

lw $t1,-12($fp)

sub $t2,$t0,$t1

bgez $t2,LABEL4

la $a0,STR40

li $v0,4

syscall

LABEL4:

li $t0,4

sll $s5,$t0,2

la $s0,v2

add $s0,$s5,$s0

lw $t7,0($s0)

sw $t7,-12($fp)

lw $t0,-8($fp)

lw $t1,-12($fp)

sub $t2,$t0,$t1

bgtz $t2,LABEL5

la $a0,STR45

li $v0,4

syscall

LABEL5:

li $t0,2

subi $sp,$sp,4

sw $t0,0($sp)

jal Fib

move $t0,$v0

sw $t0,-12($fp)

lw $t0,-8($fp)

lw $t1,-12($fp)

sub $t2,$t0,$t1

blez $t2,LABEL6

la $a0,STR51

li $v0,4

syscall

LABEL6:

lw $t0,-8($fp)

li $t1,3

sub $t2,$t0,$t1

beq $t2,$0,LABEL7

la $a0,STR55

li $v0,4

syscall

LABEL7:

lw $ra,-4($fp)

addi $sp,$fp,4

lw $fp,0($fp)

jr $ra

lw $ra,-4($fp)

addi $sp,$fp,4

lw $fp,0($fp)

jr $ra

whilest:

li $t0,0

lw $s4,0($sp)

subi $sp,$sp,8

sw $s4,0($sp)

addi $sp,$sp,12

lw $s4,0($sp)

subi $sp,$sp,8

sw $s4,0($sp)

addi $sp,$sp,12

move $s1,$fp

move $fp,$sp

addi $fp,$fp,-4

sw $s1,0($fp)

sw $ra,-4($fp)

addi $sp,$sp,-8

li $t1,5

TEP\_2:

lw $0,0($sp)

addi $sp,$sp,-4

addi $t0,$t0,1

bne $t0,$t1,TEP\_2

lw $t0,-8($fp)

sw $t0,-16($fp)

LABEL8:

lw $t0,-8($fp)

li $t1,0

sub $t2,$t0,$t1

bltz $t2,LABEL9

la $a0,STR68

li $v0,4

syscall

lw $t0,-16($fp)

lw $t1,-8($fp)

sub $t0,$t0,$t1

sw $t0,-20($fp)

lw $t0,-20($fp)

li $t1,1

add $t0,$t0,$t1

sw $t0,-24($fp)

lw $a0,-24($fp)

li $v0,1

syscall

la $a0,STR72

li $v0,4

syscall

lw $a0,-12($fp)

li $v0,11

syscall

la $a0,STR74

li $v0,4

syscall

lw $t0,-8($fp)

li $t1,1

sub $t0,$t0,$t1

sw $t0,-20($fp)

lw $t0,-20($fp)

sw $t0,-8($fp)

j LABEL8

LABEL9:

lw $ra,-4($fp)

addi $sp,$fp,4

lw $fp,0($fp)

jr $ra

lw $ra,-4($fp)

addi $sp,$fp,4

lw $fp,0($fp)

jr $ra

switchst:

li $t0,0

move $s1,$fp

move $fp,$sp

addi $fp,$fp,-4

sw $s1,0($fp)

sw $ra,-4($fp)

addi $sp,$sp,-8

li $t1,1

TEP\_3:

lw $0,0($sp)

addi $sp,$sp,-4

addi $t0,$t0,1

bne $t0,$t1,TEP\_3

la $t8,v1

lw $t0,0($t8)

li $t1,10

sub $t2,$t0,$t1

bne $t2,$0,LABEL11

la $t8,v1

lw $t0,0($t8)

li $t1,1

add $t0,$t0,$t1

sw $t0,-8($fp)

lw $t0,-8($fp)

la $t9,v1

sw $t0,0($t9)

li $a0,115

li $v0,11

syscall

la $a0,STR88

li $v0,4

syscall

j LABEL10

LABEL11:

lw $t0,-8($fp)

li $t1,8

sub $t2,$t0,$t1

bne $t2,$0,LABEL12

la $t8,v1

lw $t0,0($t8)

li $t1,2

sub $t0,$t0,$t1

sw $t0,-8($fp)

lw $t0,-8($fp)

la $t9,v1

sw $t0,0($t9)

la $t8,v1

lw $t0,0($t8)

li $t1,2

add $t0,$t0,$t1

sw $t0,-8($fp)

lw $a0,-8($fp)

li $v0,1

syscall

la $a0,STR97

li $v0,4

syscall

j LABEL10

LABEL12:

la $a0,STR100

li $v0,4

syscall

j LABEL10

LABEL10:

la $a0,STR103

li $v0,4

syscall

la $t8,v1

lw $a0,0($t8)

li $v0,1

syscall

la $a0,STR105

li $v0,4

syscall

lw $ra,-4($fp)

addi $sp,$fp,4

lw $fp,0($fp)

jr $ra

lw $ra,-4($fp)

addi $sp,$fp,4

lw $fp,0($fp)

jr $ra

main:

li $t0,0

move $s1,$fp

move $fp,$sp

addi $fp,$fp,-4

sw $s1,0($fp)

sw $ra,-4($fp)

addi $sp,$sp,-8

li $t1,11

TEP\_4:

lw $0,0($sp)

addi $sp,$sp,-4

addi $t0,$t0,1

bne $t0,$t1,TEP\_4

li $t0,0

sw $t0,-8($fp)

LABEL13:

lw $t0,-8($fp)

li $t1,13

sub $t2,$t0,$t1

beq $t2,$0,LABEL14

lw $t0,-8($fp)

subi $sp,$sp,4

sw $t0,0($sp)

jal Fib

move $t0,$v0

sw $t0,-32($fp)

lw $t3,-32($fp)

lw $t4,-8($fp)

sll $s5,$t4,2

la $s6,v2

add $s6,$s6,$s5

sw $t3,0($s6)

lw $t0,-8($fp)

li $t1,1

add $t0,$t0,$t1

sw $t0,-32($fp)

lw $t0,-32($fp)

sw $t0,-8($fp)

j LABEL13

LABEL14:

li $v0,5

syscall

move $t0,$v0

sw $t0,-12($fp)

lw $t0,-12($fp)

subi $sp,$sp,4

sw $t0,0($sp)

jal ifst

li $v0,5

syscall

move $t0,$v0

sw $t0,-16($fp)

li $v0,12

syscall

move $t0,$v0

sw $t0,-20($fp)

lw $t0,-16($fp)

subi $sp,$sp,4

sw $t0,0($sp)

lw $t0,-20($fp)

subi $sp,$sp,4

sw $t0,0($sp)

jal whilest

li $t0,6

li $t1,4

mult $t0,$t1

mflo $t0

sw $t0,-32($fp)

lw $t0,-32($fp)

li $t1,8

div $t0,$t1

mflo $t0

sw $t0,-36($fp)

lw $t0,-12($fp)

lw $t1,-36($fp)

add $t0,$t0,$t1

sw $t0,-40($fp)

lw $t0,-40($fp)

li $t1,1

sub $t0,$t0,$t1

sw $t0,-44($fp)

lw $t0,-44($fp)

li $t1,2

add $t0,$t0,$t1

sw $t0,-48($fp)

lw $t0,-48($fp)

la $t9,v1

sw $t0,0($t9)

la $t8,v1

lw $t0,0($t8)

sll $s5,$t0,2

la $s0,v2

add $s0,$s5,$s0

lw $t7,0($s0)

sw $t7,-32($fp)

lw $a0,-32($fp)

li $v0,1

syscall

la $a0,STR143

li $v0,4

syscall

jal switchst

li $v0,10

syscall

li $v0,10

syscall

测试数据1：6 5 e

输出结果：

This number is great than or equal to1.

This number is smaller than 12.

This number is great than 2.

This number is not equal to c1.

count:1

e

count:2

e

count:3

e

count:4

e

count:5

e

count:6

e

89

s

v1 is equal to 11

测试数据2：3 8 &

输出结果：This number is great than or equal to1.

This number is smaller than 12.

This number is smaller than or equal to 5.

This number is great than 2.

count:1

&

count:2

&

count:3

&

count:4

&

count:5

&

count:6

&

count:7

&

count:8

&

count:9

&

21

v1 does not equal 10 or equal to 8;

v1 is equal to 7

测试数据3：8 12 \*

测试结果：

This number is great than or equal to1.

This number is smaller than 12.

This number is great than 2.

This number is not equal to c1.

count:1

\*

count:2

\*

count:3

\*

count:4

\*

count:5

\*

count:6

\*

count:7

\*

count:8

\*

count:9

\*

count:10

\*

count:11

\*

count:12

\*

count:13

\*

233

v1 does not equal 10 or equal to 8;

v1 is equal to 12

三次测试结果均与预期结果一致。