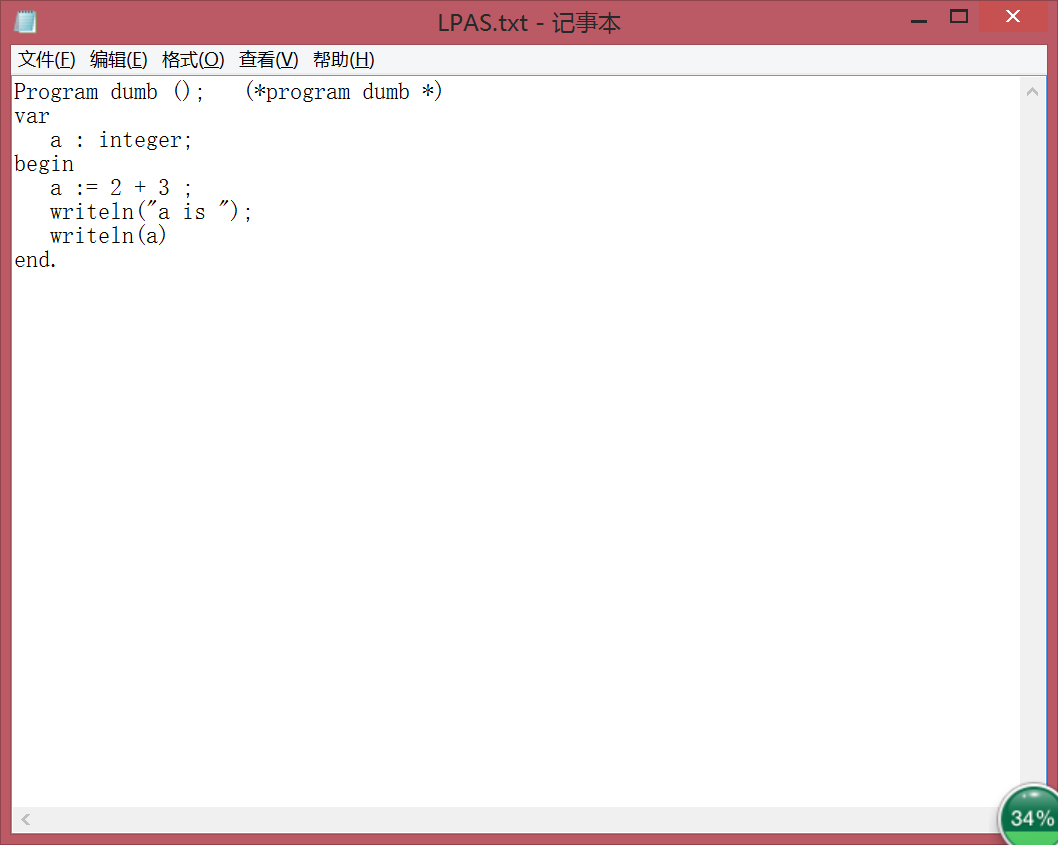
Test

Assign statement, wrireln a string and integer, addition on integer.



li,$t0,2

sw,$t0,-4($fp)

li,$t0,3

sw,$t0,-8($fp)

lw,$t0,-4($sp)

lw,$t1,-8($sp)

add,$t0 $t0 $t1

sw,$t0,-12($sp)

lw,$t0,-12($fp)

sw,$t0,0($fp)

.data

StrLabel1: .asciiz"a is "

.text

la $a0 StrLabel1

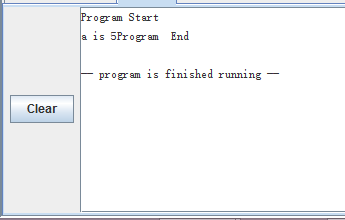
li $v0 4

syscall

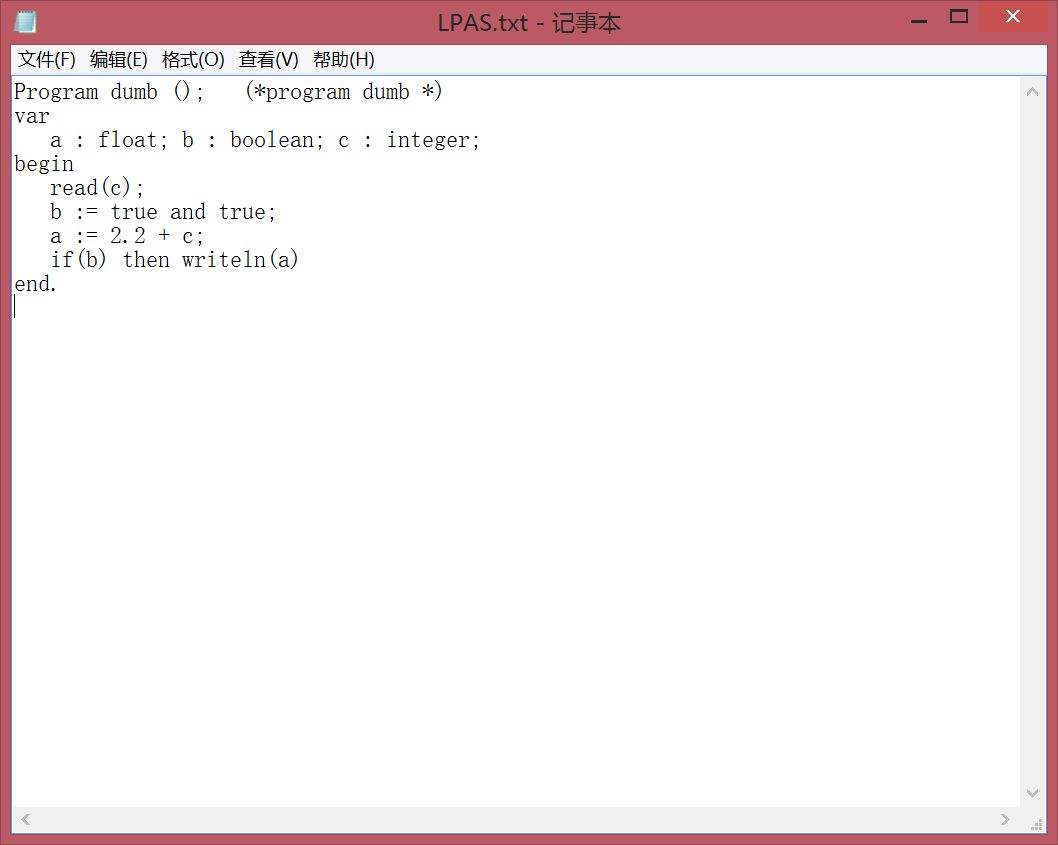
lw $a0 0($fp)

li $v0 1

syscall



Float addition、boolean assignmenet、boolean(and )、if statement 、writeln float、read float:



li $v0 6

syscall

s.s $f0,-8($fp)

li,$t0,1

sw,$t0,-12($fp)

li,$t0,1

sw,$t0,-16($fp)

lw,$t0,-12($sp)

lw,$t1,-16($sp)

and,$t0 $t0 $t1

sw,$t0,-20($sp)

l.s $f1 StrLabel1

s.s $f1 -24($fp)

l.s $f1,-24($sp)

l.s $f2,-8($sp)

add.s $f1 $f2 $f1

s.s $f1,-28($sp)

l.s $f1 -28($fp)

s.s $f1 0($fp)

beq $t0 $0 StrLabel2

l.s $f12 0($fp)

li $v0 2

syscall

StrLabel2:

#Postlog:

la $a0 ProgEnd

li $v0 4

syscall

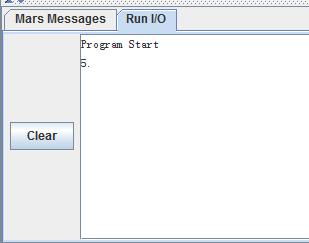
li $v0 10

syscall

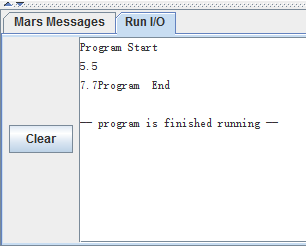
.data

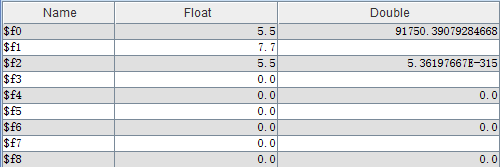
StrLabel1: .float 2.2

Input 5.5

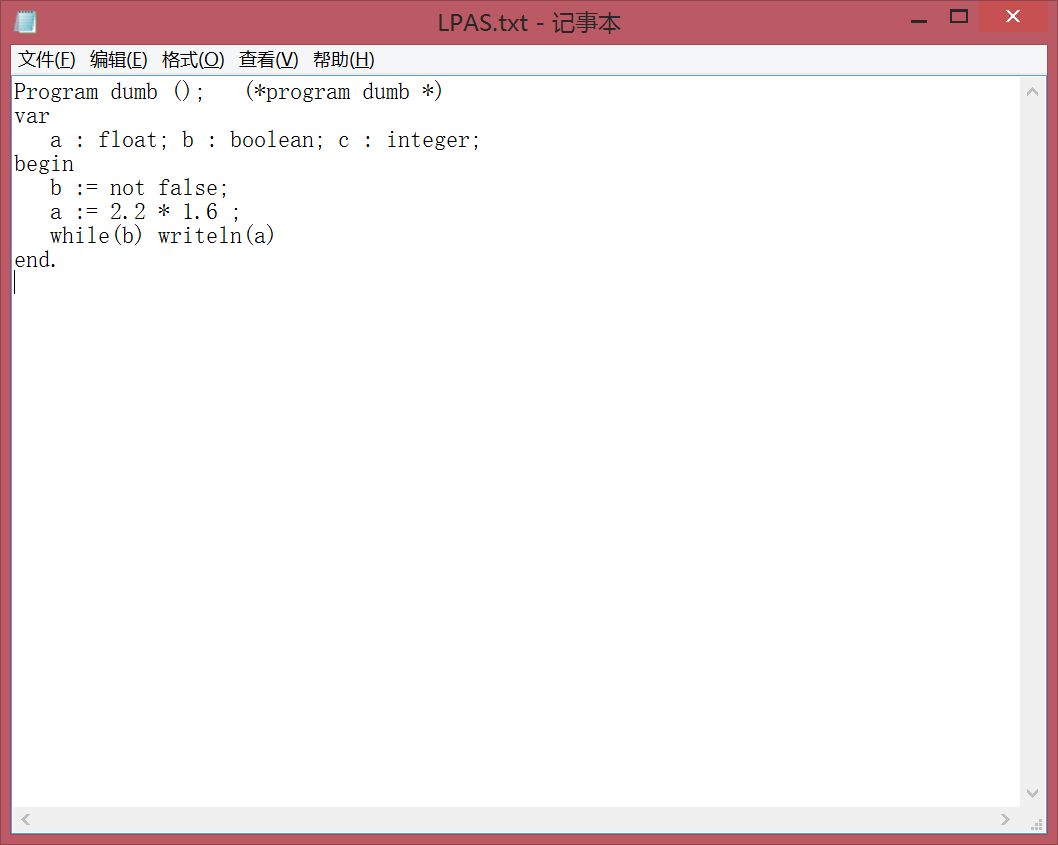


Read(c), so assign 5.5 to c, and then make an addition of 2.2 and 5(->c), and the result is 7.7.





Boolean(not)，while statement, float multiplication, float assignment:



li,$t0,0

sw,$t0,-12($fp)

not $t0 $t0

l.s $f1 StrLabel1

s.s $f1 -16($fp)

l.s $f1 StrLabel2

s.s $f1 -20($fp)

l.s $f1,-16($sp)

l.s $f2,-20($sp)

mul.s $f1 $f2 $f1

s.s $f1 -24($sp)

l.s $f1 -24($fp)

s.s $f1 0($fp)

StrLabel3:

beq $t0 $0 StrLabel4

l.s $f12 0($fp)

li $v0 2

syscall

j StrLabel3

StrLabel4:

#Postlog:

la $a0 ProgEnd

li $v0 4

syscall

li $v0 10

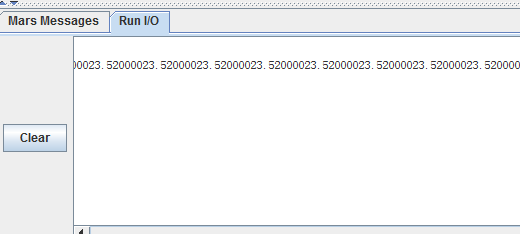
syscall

.data

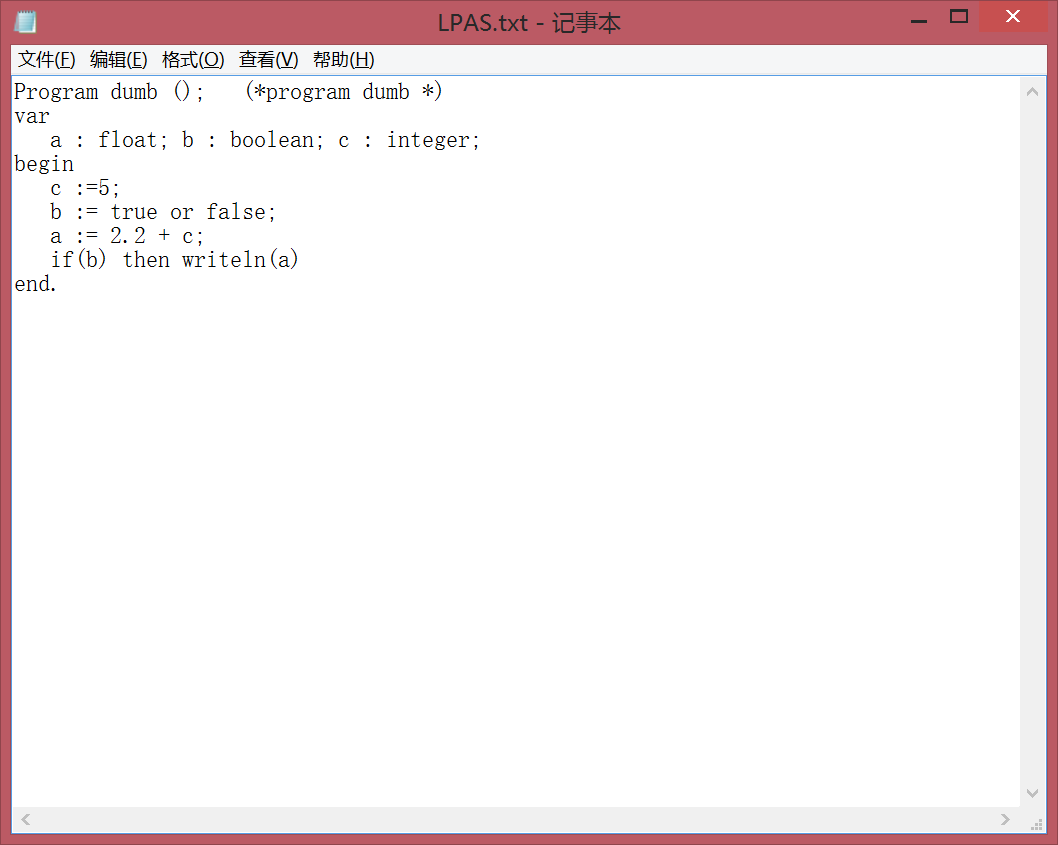
StrLabel1: .float 2.2

StrLabel2: .float 1.6

The result is 3.5200002, because I don’t have a space here, so, it looks like .52000023



Boolean(or), addition of float and integer:



li,$t0,5

sw,$t0,-12($fp)

lw,$t0,-12($fp)

sw,$t0,-8($fp)

li,$t0,1

sw,$t0,-16($fp)

li,$t0,0

sw,$t0,-20($fp)

lw,$t0,-16($sp)

lw,$t1,-20($sp)

or,$t0 $t0 $t1

sw,$t0,-24($sp)

l.s $f1 StrLabel1

s.s $f1 -28($fp)

l.s $f1,-28($sp)

lwc1 $f0 -8($sp)

cvt.s.w $f2 $f0

add.s $f1 $f2 $f1

s.s $f1,-32($sp)

l.s $f1 -32($fp)

s.s $f1 0($fp)

beq $t0 $0 StrLabel2

l.s $f12 0($fp)

li $v0 2

syscall

StrLabel2:

#Postlog:

la $a0 ProgEnd

li $v0 4

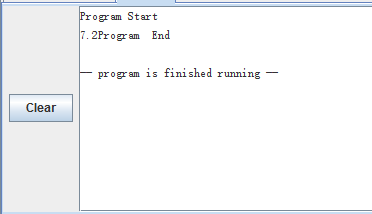
syscall

li $v0 10

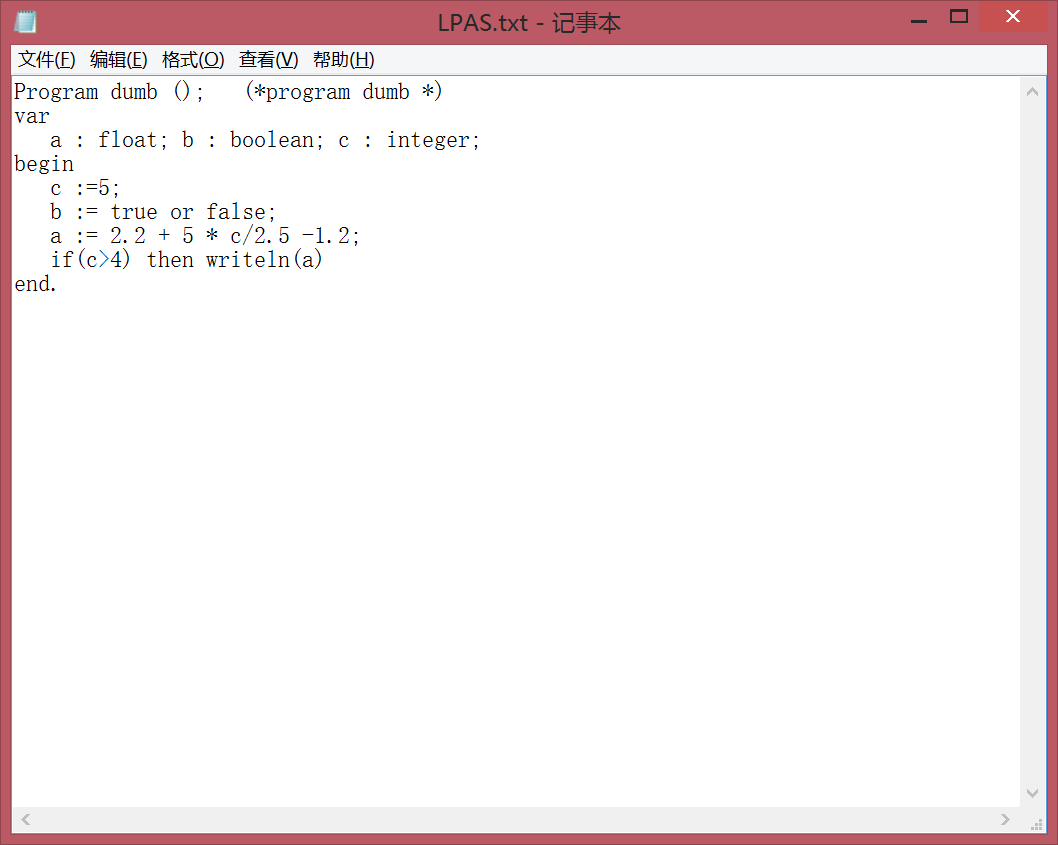
syscall

.data

StrLabel1: .float 2.2



Realation Token, complicated arithmetic of float and integer:



li,$t0,5

sw,$t0,-12($fp)

lw,$t0,-12($fp)

sw,$t0,-8($fp)

li,$t0,1

sw,$t0,-16($fp)

li,$t0,0

sw,$t0,-20($fp)

lw,$t0,-16($sp)

lw,$t1,-20($sp)

or,$t0 $t0 $t1

sw,$t0,-24($sp)

l.s $f1 StrLabel1

s.s $f1 -28($fp)

li,$t0,5

sw,$t0,-32($fp)

lw,$t0,-32($sp)

lw,$t1,-8($sp)

mul,$t0 $t0 $t1

sw,$t0,-36($sp)

l.s $f1 StrLabel2

s.s $f1 -40($fp)

l.s $f1,-40($sp)

lwc1 $f0 -36($sp)

cvt.s.w $f2 $f0

div.s $f1 $f2 $f1

s.s $f1 -44($sp)

l.s $f1,-28($sp)

l.s $f2,-44($sp)

add.s $f1 $f2 $f1

s.s $f1,-48($sp)

l.s $f1 StrLabel3

s.s $f1 -52($fp)

l.s $f1,-48($sp)

l.s $f2,-52($sp)

sub.s $f1 $f2 $f1

s.s $f1,-56($sp)

l.s $f1 -56($fp)

s.s $f1 0($fp)

li,$t0,4

sw,$t0,-60($fp)

lw,$t0,-8($sp)

lw,$t1,-60($sp)

bgt $t0 $t1 StrLabel4

li,$t0,0

sw,$t0,-64($fp)

StrLabel4:

li,$t0,1

sw,$t0,-64($fp)

beq $t0 $0 StrLabel5

l.s $f12 0($fp)

li $v0 2

syscall

StrLabel5:

#Postlog:

la $a0 ProgEnd

li $v0 4

syscall

li $v0 10

syscall

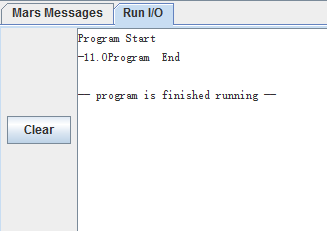
.data

StrLabel1: .float 2.2

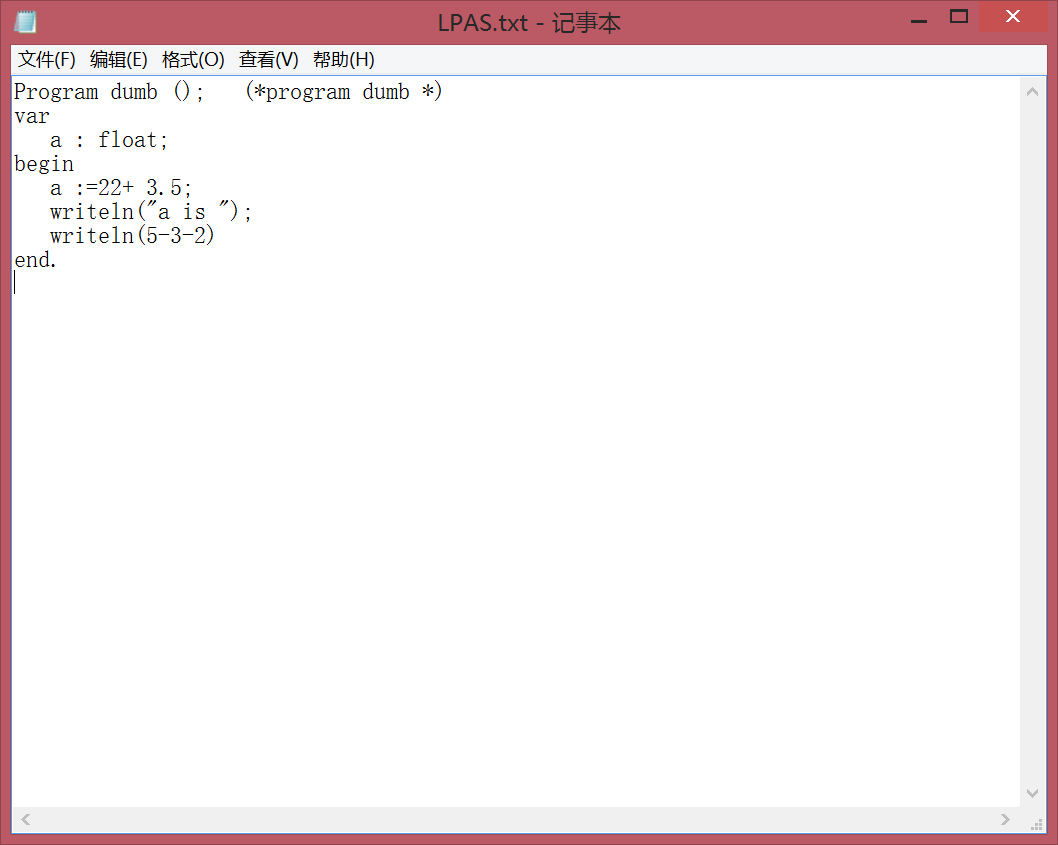
StrLabel2: .float 2.5

StrLabel3: .float 1.2

Because c equals 5, so c is bigger than 4, a is the addition of 2.2 + 5 \* c(->5)/2.5 -1.2, so a is 11.0.



Writeln(5-3-2)



#Prolog:

.text

.globl main

main:

move $fp $sp

la $a0 ProgStart

li $v0 4

syscall

#End of Prolog

li,$t0,22

sw,$t0,-4($fp)

l.s $f1 StrLabel1

s.s $f1 -8($fp)

l.s $f1,-8($sp)

lwc1 $f0 -4($sp)

cvt.s.w $f2 $f0

add.s $f1 $f2 $f1

s.s $f1,-12($sp)

l.s $f1 -12($fp)

s.s $f1 0($fp)

.data

StrLabel2: .asciiz"a is "

.text

la $a0 StrLabel2

li $v0 4

syscall

li,$t0,5

sw,$t0,-16($fp)

li,$t0,3

sw,$t0,-20($fp)

lw,$t0,-16($sp)

lw,$t1,-20($sp)

sub,$t0 $t0 $t1

sw,$t0,-24($sp)

li,$t0,2

sw,$t0,-28($fp)

lw,$t0,-24($sp)

lw,$t1,-28($sp)

sub,$t0 $t0 $t1

sw,$t0,-32($sp)

lw $a0 -32($fp)

li $v0 1

syscall

#Postlog:

la $a0 ProgEnd

li $v0 4

syscall

li $v0 10

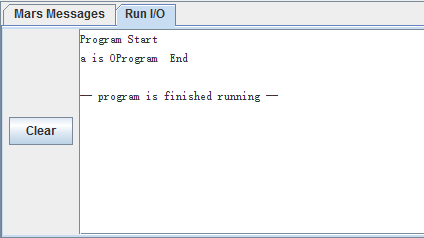
syscall

.data

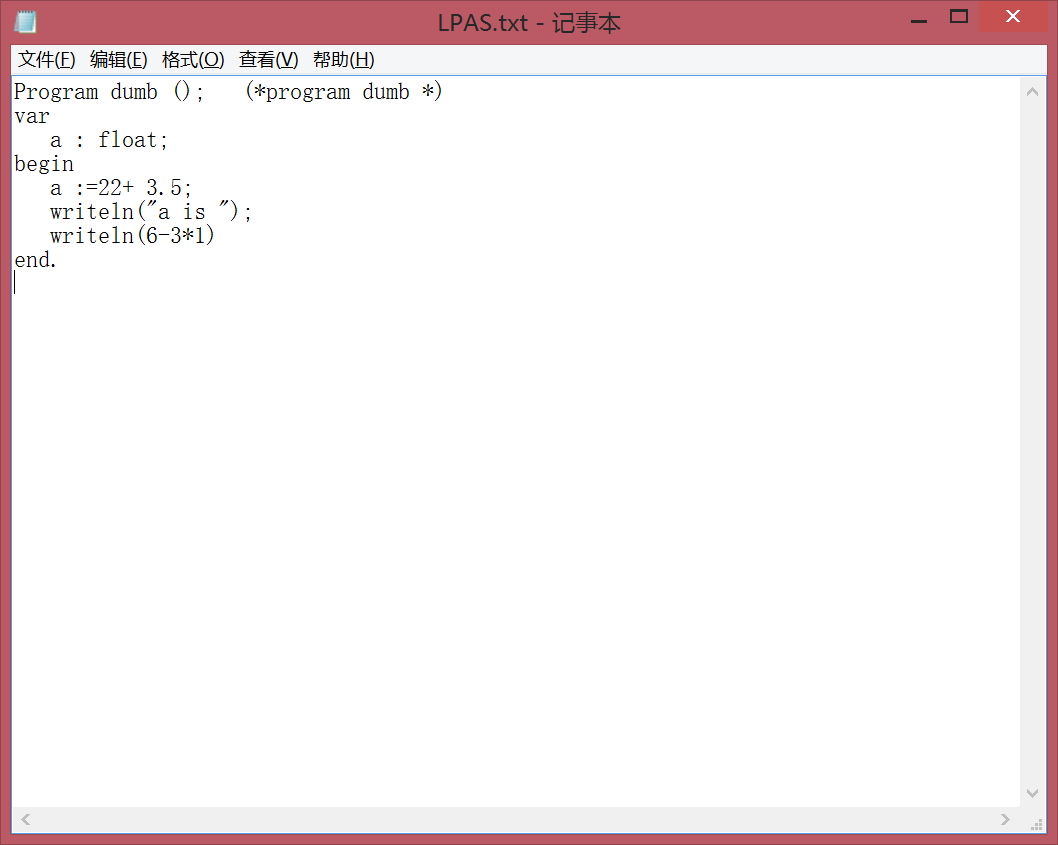
StrLabel1: .float 3.5

ProgStart: .asciiz "Program Start\n"

ProgEnd: .asciiz "Program End\n"



Writeln(6-3\*1)



#Prolog:

.text

.globl main

main:

move $fp $sp

la $a0 ProgStart

li $v0 4

syscall

#End of Prolog

li,$t0,22

sw,$t0,-4($fp)

l.s $f1 StrLabel1

s.s $f1 -8($fp)

l.s $f1,-8($sp)

lwc1 $f0 -4($sp)

cvt.s.w $f2 $f0

add.s $f1 $f2 $f1

s.s $f1,-12($sp)

l.s $f1 -12($fp)

s.s $f1 0($fp)

.data

StrLabel2: .asciiz"a is "

.text

la $a0 StrLabel2

li $v0 4

syscall

li,$t0,6

sw,$t0,-16($fp)

li,$t0,3

sw,$t0,-20($fp)

li,$t0,1

sw,$t0,-24($fp)

lw,$t0,-20($sp)

lw,$t1,-24($sp)

mul,$t0 $t0 $t1

sw,$t0,-28($sp)

lw,$t0,-16($sp)

lw,$t1,-28($sp)

sub,$t0 $t0 $t1

sw,$t0,-32($sp)

lw $a0 -32($fp)

li $v0 1

syscall

#Postlog:

la $a0 ProgEnd

li $v0 4

syscall

li $v0 10

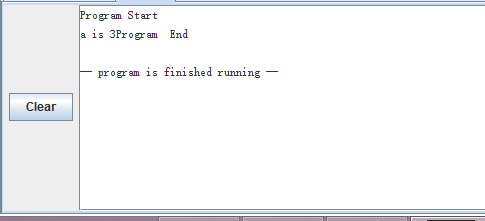
syscall

.data

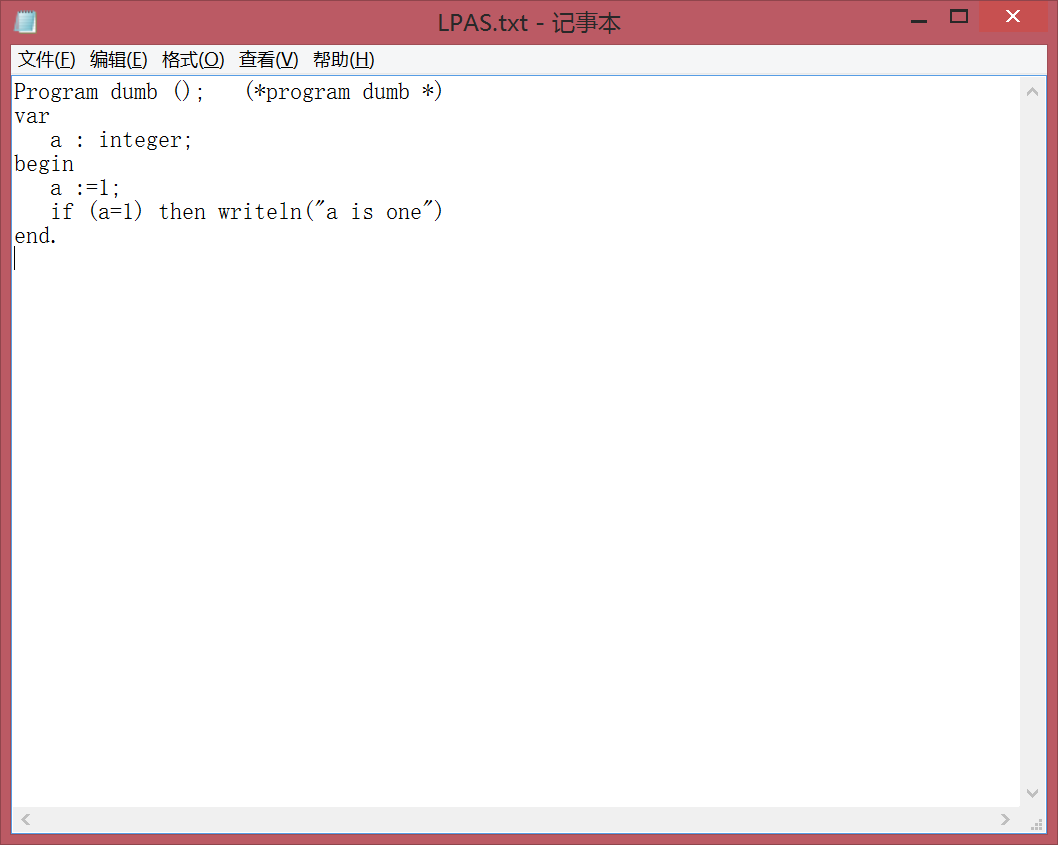
StrLabel1: .float 3.5

ProgStart: .asciiz "Program Start\n"

ProgEnd: .asciiz "Program End\n"



If (a=1) then writeln(“a is one”)



#Prolog:

.text

.globl main

main:

move $fp $sp

la $a0 ProgStart

li $v0 4

syscall

#End of Prolog

li,$t0,1

sw,$t0,-4($fp)

lw,$t0,-4($fp)

sw,$t0,0($fp)

li,$t0,1

sw,$t0,-8($fp)

lw,$t0,0($sp)

lw,$t1,-8($sp)

beq $t0 $t1 StrLabel1

li,$t0,0

sw,$t0,-12($fp)

StrLabel1:

li,$t0,1

sw,$t0,-12($fp)

beq $t0 $0 StrLabel2

.data

StrLabel3: .asciiz"a is one"

.text

la $a0 StrLabel3

li $v0 4

syscall

StrLabel2:

#Postlog:

la $a0 ProgEnd

li $v0 4

syscall

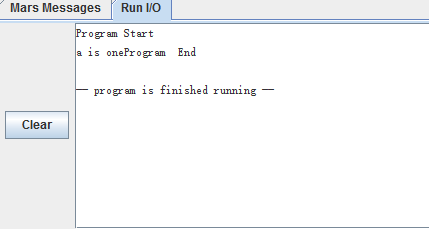
li $v0 10

syscall

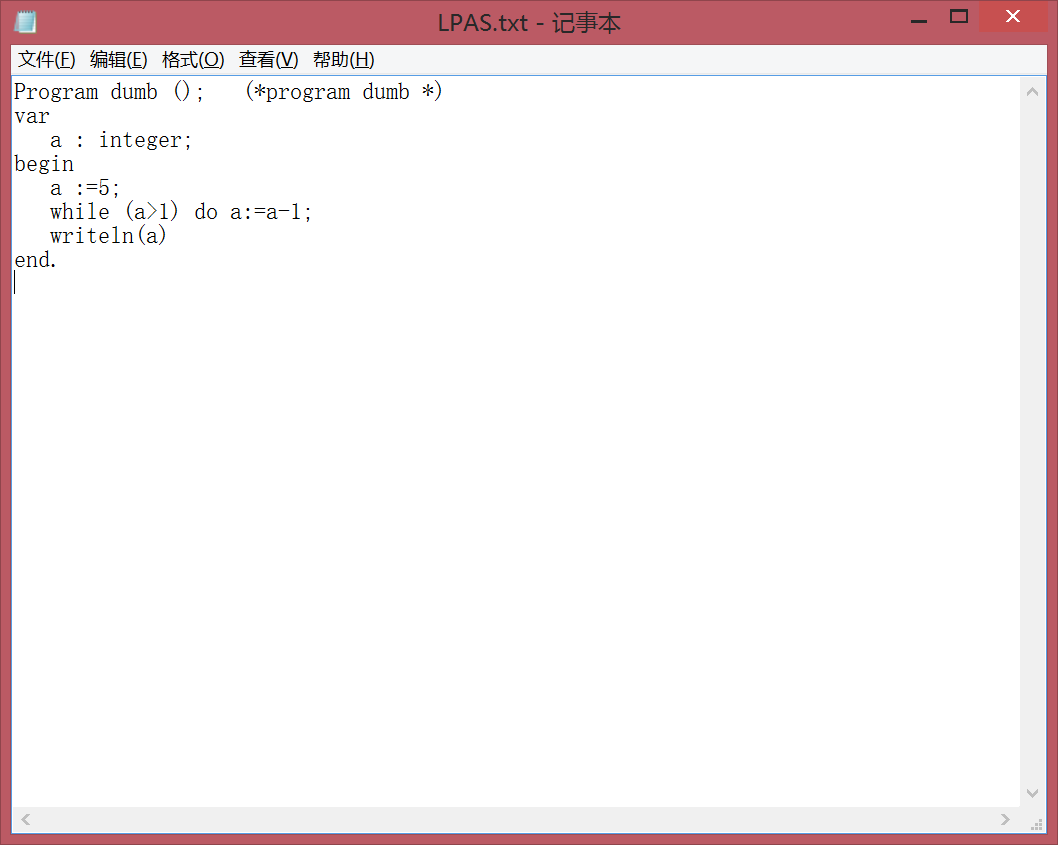
.data

ProgStart: .asciiz "Program Start\n"

ProgEnd: .asciiz "Program End\n"



while (a>1) do a:=a-1;



#Prolog:

.text

.globl main

main:

move $fp $sp

la $a0 ProgStart

li $v0 4

syscall

#End of Prolog

li,$t0,5

sw,$t0,-4($fp)

lw,$t0,-4($fp)

sw,$t0,0($fp)

StrLabel1:

li,$t0,1

sw,$t0,-8($fp)

lw,$t0,0($sp)

lw,$t1,-8($sp)

bgt $t0 $t1 StrLabel2

li,$t0,0

sw,$t0,-12($fp)

StrLabel2:

beq $t0 $0 StrLabel3

li,$t0,1

sw,$t0,-16($fp)

lw,$t0,0($sp)

lw,$t1,-16($sp)

sub,$t0 $t0 $t1

sw,$t0,-20($sp)

lw,$t0,-20($fp)

sw,$t0,0($fp)

j StrLabel1

StrLabel3:

lw $a0 0($fp)

li $v0 1

syscall

#Postlog:

la $a0 ProgEnd

li $v0 4

syscall

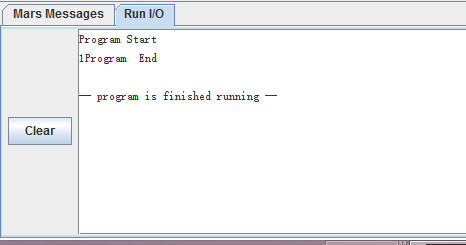
li $v0 10

syscall

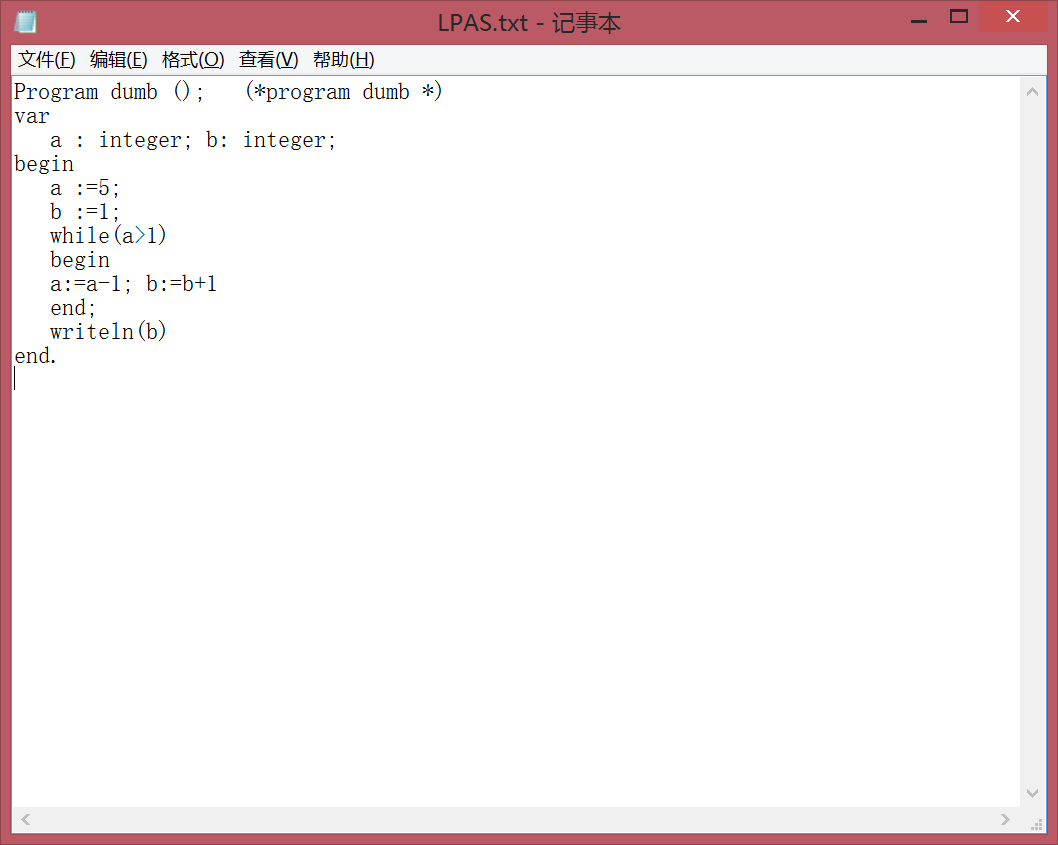
.data

ProgStart: .asciiz "Program Start\n"

ProgEnd: .asciiz "Program End\n"



a :=5; b :=1; while(a>1) begin a:=a-1; b:=b+1 end; writeln(b)



#Prolog:

.text

.globl main

main:

move $fp $sp

la $a0 ProgStart

li $v0 4

syscall

#End of Prolog

li,$t0,5

sw,$t0,-8($fp)

lw,$t0,-8($fp)

sw,$t0,0($fp)

li,$t0,1

sw,$t0,-12($fp)

lw,$t0,-12($fp)

sw,$t0,-4($fp)

StrLabel1:

li,$t0,1

sw,$t0,-16($fp)

lw,$t0,0($sp)

lw,$t1,-16($sp)

bgt $t0 $t1 StrLabel2

li,$t0,0

sw,$t0,-20($fp)

StrLabel2:

beq $t0 $0 StrLabel3

li,$t0,1

sw,$t0,-24($fp)

lw,$t0,0($sp)

lw,$t1,-24($sp)

sub,$t0 $t0 $t1

sw,$t0,-28($sp)

lw,$t0,-28($fp)

sw,$t0,0($fp)

li,$t0,1

sw,$t0,-32($fp)

lw,$t0,-4($sp)

lw,$t1,-32($sp)

add,$t0 $t0 $t1

sw,$t0,-36($sp)

lw,$t0,-36($fp)

sw,$t0,-4($fp)

j StrLabel1

StrLabel3:

lw $a0 -4($fp)

li $v0 1

syscall

#Postlog:

la $a0 ProgEnd

li $v0 4

syscall

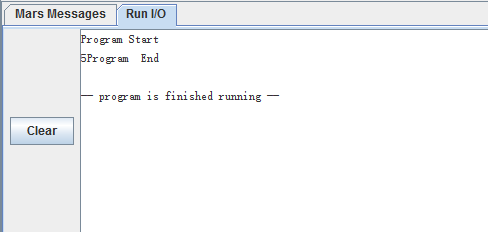
li $v0 10

syscall

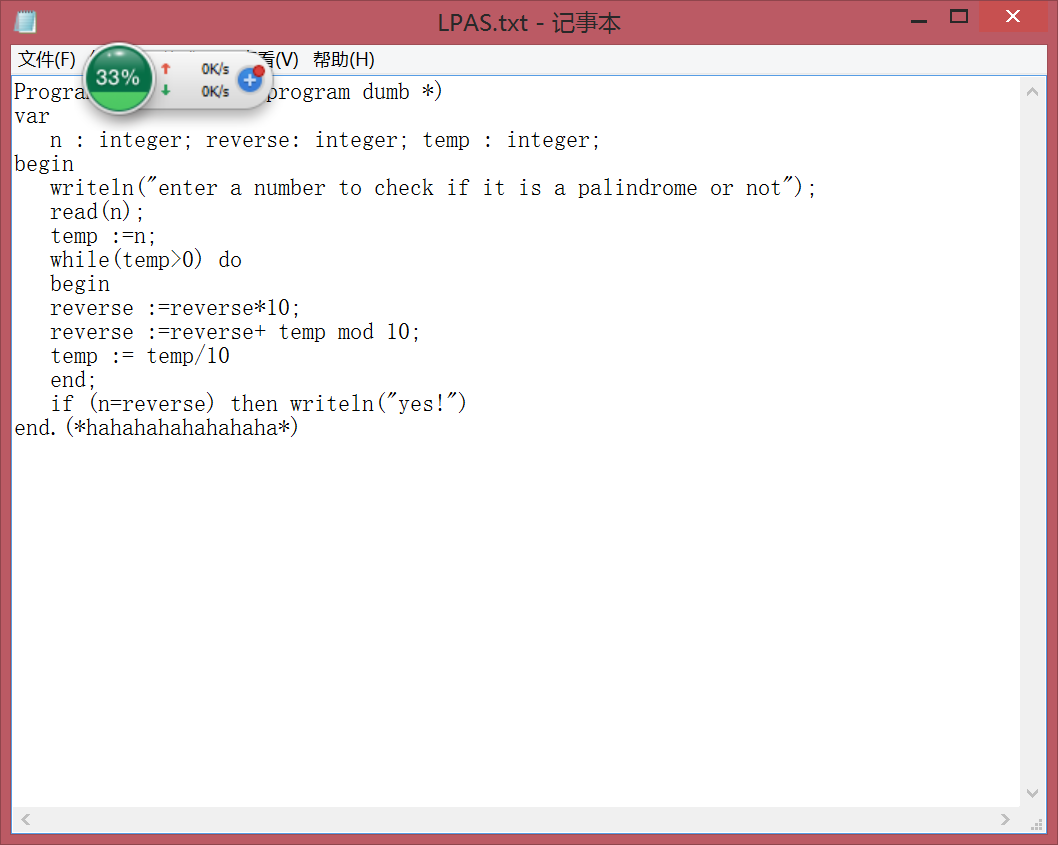
.data

ProgStart: .asciiz "Program Start\n"

ProgEnd: .asciiz "Program End\n"



Check palindrome (also test a comment after end.)



#Prolog:

.text

.globl main

main:

move $fp $sp

la $a0 ProgStart

li $v0 4

syscall

#End of Prolog

.data

StrLabel1: .asciiz"enter a number to check if it is a palindrome or not"

.text

la $a0 StrLabel1

li $v0 4

syscall

li $v0 5

syscall

sw,$v0,0($fp)

lw,$t0,0($fp)

sw,$t0,-8($fp)

StrLabel2:

li,$t0,0

sw,$t0,-12($fp)

lw,$t0,-8($sp)

lw,$t1,-12($sp)

bgt $t0 $t1 StrLabel3

li,$t0,0

sw,$t0,-16($fp)

StrLabel3:

beq $t0 $0 StrLabel4

li,$t0,10

sw,$t0,-20($fp)

lw,$t0,-4($sp)

lw,$t1,-20($sp)

mul,$t0 $t0 $t1

sw,$t0,-24($sp)

lw,$t0,-24($fp)

sw,$t0,-4($fp)

li,$t0,10

sw,$t0,-28($fp)

lw,$t0,-8($sp)

lw,$t1,-28($sp)

div,$t0 $t0 $t1

mfhi $t0

sw,$t0,-32($sp)

lw,$t0,-4($sp)

lw,$t1,-32($sp)

add,$t0 $t0 $t1

sw,$t0,-36($sp)

lw,$t0,-36($fp)

sw,$t0,-4($fp)

li,$t0,10

sw,$t0,-40($fp)

lw,$t0,-8($sp)

lw,$t1,-40($sp)

div,$t0 $t0 $t1

sw,$t0,-44($sp)

lw,$t0,-44($fp)

sw,$t0,-8($fp)

j StrLabel2

StrLabel4:

lw,$t0,0($sp)

lw,$t1,-4($sp)

beq $t0 $t1 StrLabel5

li,$t0,0

sw,$t0,-20($fp)

StrLabel5:

beq $t0 $0 StrLabel6

.data

StrLabel7: .asciiz"yes!"

.text

la $a0 StrLabel7

li $v0 4

syscall

StrLabel6:

#Postlog:

la $a0 ProgEnd

li $v0 4

syscall

li $v0 10

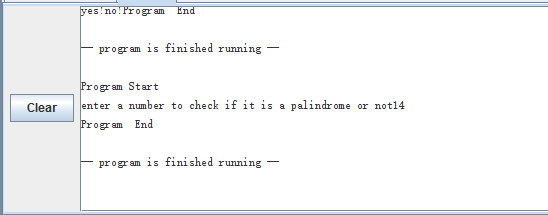
syscall

.data

ProgStart: .asciiz "Program Start\n"

ProgEnd: .asciiz "Program End\n"

First we input 14



Then we input 12344321

