作业 9：第 10 次课课后练习

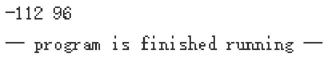
6.1

```
.data
N:  .word  8
X:  .word  -112,5,4,8,12,-5,96,-88
sp:  .asciiz  " "
enter:  .asciiz  "\n"
.globl  main
.text
main:
la  $8,N
la  $9,X
lw  $8,($8)
addiu  $sp,$sp,-20
sw  $8,($sp)
sw  $9,4($sp)
sw  $ra,16($sp)
jal  MinMax
lw  $a0,8($sp)
li  $v0,1
syscall
la  $a0,sp
li  $v0,4
syscall
lw  $a0,12($sp)
li  $v0,1
syscall
addiu  $sp,$sp,20
li  $v0,10
syscall
MinMax:
lw  $8,($sp)  #n
lw  $9,4($sp)  #&X
lw  $10,($9)  #min
lw  $11,($9)  #max
addi  $9,$9,4  #i
addi  $8,$8,-1
loop:
lw  $12,($9)  #x[i]
blt  $12,$10,ifmin
```

```
bgt  $12,$11,ifmax
ifend:
addi  $9,$9,4
addi  $8,$8,-1
bgtz  $8,loop
sw  $10,8($sp)
sw  $11,12($sp)
jr  $ra
ifmax:
lw  $11,($9)
b  ifend
ifmin:
lw  $10,($9)
b  ifend
```

结果:

输出[-112, 5, 4, 8, 12, -5, 96, -88]的最小值和最大值



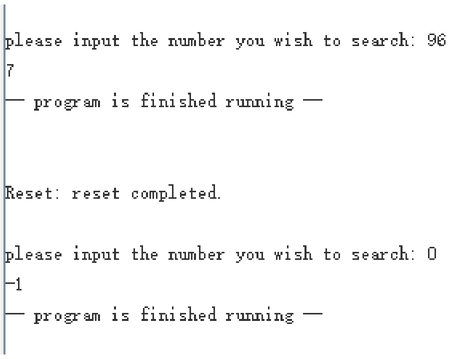
```
-112 96
-- program is finished running --
```

6.2

```
.data
N:  .word  8
X:  .byte  -18,5,4,8,12,-5,96,-8
sp:  .asciiz  " "
enter:  .asciiz  "\n"
tip:  .asciiz  "please input the number you wish to search: "
.globl  main
.text
main:
la  $8,N
la  $9,X
la  $a0,tip
li  $v0,4
syscall
li  $v0,5
syscall
lw  $8,($8)
addiu  $sp,$sp,-20
sw  $8,($sp)
sw  $9,4($sp)
sw   $v0,8($sp)
sw   $ra,16($sp)
jal  Search
lw   $a0,12($sp)
li   $v0,1
syscall
addiu  $sp,$sp,20
li   $v0,10
syscall
Search:
lw   $8,($sp)   #n
lw   $9,4($sp)   #&X
lw   $10,8($sp)   #V
li   $11,-1   #return
li   $13,1   #return
loop:
lb   $12,($9)   #x[i]
beq     $12,$10,setindex
addi   $9,$9,1
addi   $8,$8,-1
addi   $13,$13,1
bgtz   $8,loop
loopend:
sw   $11,12($sp)
jr   $ra
setindex:
move   $11,$13
b   loopend
```

结果:

输出数组[-18， 5， 4，8，12，-5，96，-8]中所查找数值的位置



```
please input the number you wish to search: 96
7
-- program is finished running --


Reset: reset completed.

please input the number you wish to search: 0
-1
-- program is finished running --
```

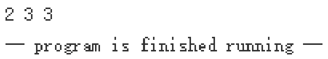
6.3

```
.data
N:  .word  8
X:  .asciiz  "a123vdAQ"
sp:  .asciiz  " "
enter:  .asciiz  "\n"
.globl  main
.text
main:
la  $8,N
la  $9,X
lw  $8,($8)
addiu  $sp,$sp,-24
sw  $8,($sp)
sw  $9,4($sp)
sw  $ra,16($sp)
jal  Scan
lw  $a0,8($sp)
jal  print
lw  $a0,12($sp)
jal  print
lw  $a0,16($sp)
jal  print
addiu  $sp,$sp,24
li  $v0,10
syscall
Scan:
lw  $8,($sp)  #n
lw  $9,4($sp)  #&X
li  $10,0  #大写字母个数U
li  $11,0  #小写字母个数L
li  $12,0  #十进制数字个数D
loop:
lb  $13,($9)  #x[i]
addi  $13,$13,-48
bltz  $13,end
addi  $13,$13,-9
blez  $13,calD
addi  $13,$13,-8
bltz  $13,end
addi  $13,$13,-25
blez  $13,calU
addi  $13,$13,-7
bltz  $13,end
addi  $13,$13,-25
```

```
blez  $13,calL
calU:
addi  $10,$10,1
b  end
calL:
addi  $11,$11,1
b  end
calD:
addi  $12,$12,1
b  end
end:
addi  $9,$9,1
addi  $8,$8,-1
bgtz  $8,loop
sw  $10,8($sp)
sw  $11,12($sp)
sw  $12,16($sp)
jr  $ra
print:
li  $v0,1
syscall
la  $a0,sp
li  $v0,4
syscall
jr  $ra
```

结果:

输出数组[‘a’, ‘1’, ‘2’, ‘3’, ‘v’, ‘d’, ‘a’, ’A’, ’Q’]中大写字母、小写字母、十进制数字的个数



```
2 3 3
— program is finished running —
```

6.9

```
.data
data:  .word  1,2,1
sp:  .asciiz  "  "
enter:  .asciiz  "\n"
err:  .asciiz  "calculate  error!!!"
.globl  main
.text
main:
la  $8,data
```

```
lw  $9,0($8)
lw  $10,4($8)
lw  $11,8($8)
addiu  $sp,$sp,-28
sw  $9,($sp)
sw  $10,4($sp)
sw  $11,8($sp)
sw  $ra,24($sp)
jal  Roots
lw  $8,12($sp)
lw  $9,16($sp)
lw  $10,20($sp)
addiu  $sp,$sp,28
#  打印结果
move  $a0,$8
jal  print
addi  $8,$8,-3
beqz  $8,error
move  $a0,$20
jal  print
move  $a0,$21
jal  print
#  end
exit:
li  $v0,10
syscall
Roots:
lw  $16,($sp)  #a
lw  $17,4($sp)  #b
lw  $18,8($sp)  #c
li  $19,0  #status
li  $20,0  #r1
li  $21,0  #r2
mul  $14,$17,$17  #b*b
mul  $15,$16,$18  #a*c
sll  $15,$15,2  #4*a*c
sub  $14,$14,$15  #  b*b-4*a*c
beqz  $14,type1
#  求|b^2-4ac|的开根号
abs  $22,$14  #  |b^2-4ac|
addiu  $sp,$sp,-12
sw  $22,($sp)
sw  $ra,8($sp)
jal  sqrt
```

```
lw  $22,4($sp)
lw  $ra,8($sp)
addiu  $sp,$sp,12
#  end
bgtz  $14,type0
b  type2
end:
sw  $19,12($sp)
sw  $20,16($sp)
sw  $21,20($sp)
jr  $ra
type0:
beqz  $22,type3
sub  $8,$0,$17
sub  $9,$8,$22
add  $10,$8,$22
div  $20,$9,$16
sra  $20,$20,1
div  $21,$10,$16
sra  $21,$21,1
b  end
type1:
ori  $19,$19,1
beqz  $17,type3
sub  $20,$0,$17
sll  $8,$16,1
div  $20,$20,$8
move  $21,$20
b  end
type2:
ori  $19,$19,2
sub  $20,$0,$17
div  $20,$20,$16
sra  $20,$20,1
div  $21,$22,$16
sra  $21,$21,1
b  end
type3:
ori  $19,$19,3
b  end
print:
li  $v0,1
syscall
la  $a0,sp
```

```
li  $v0,4
syscall
jr  $ra
sqrt:
lw  $8,($sp)
li  $9,1
loop:
mul  $10,$9,$9
blt  $10,$8,next
beq  $10,$8,end_
li  $9,0
end_:
sw  $9,4($sp)
jr  $ra
next:
addi,$9,$9,1
b  loop
error:
la  $a0,err
li  $v0,4
syscall
b  exit
```

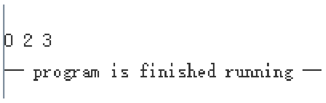
结果：输出 Status,R1,R2(当有两个复数根时，对应 R1+i*R2)

$x^2-5x+6=0$

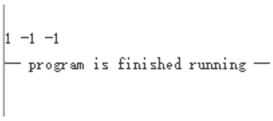
$x^2+2x+1=0$

$x^2+4x+8=0$

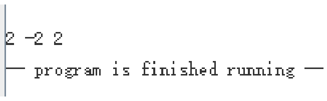
$x^2+1=0$



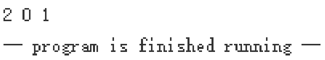
```
0 2 3
— program is finished running —
```



```
1 -1 -1
— program is finished running —
```



```
2 -2 2
— program is finished running —
```



```
2 0 1
— program is finished running —
```

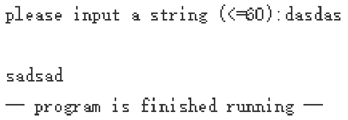
作业 10：第 11 次课课后练习

6.10

```
.data
tip: .asciiz ″please input a string (<=60):″
str: .space 60
.globl main
.text
main:
li $v0,4
la $a0,tip
syscall
li $v0,8
la $a0,str
li $a1,60
syscall
jal Reverse
li $v0,10
syscall
Reverse:
addiu $sp,$sp,-60
move $a1,$sp
addi $a1,$a1,60
sb $0,($a1)
move $8,$a0
loop:
lb $9,($8)
beqz $9,end
addi $a1,$a1,-1
sb $9,($a1)
addi $8,$8,1
b loop
end:
# 打印
move $a0,$a1
li $v0,4
syscall
addi $sp,$sp,60
jr $ra

```

结果:



```
please input a string (<=60):dasdas

sadsad
— program is finished running —
```

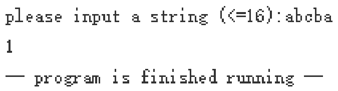
6.11

```
.data
tip:  .asciiz  "please  input  a  string  (<=16):"
str:  .space  60
.globl  main
.text
main:
li  $v0,4
la  $a0,tip
syscall
li  $v0,8
la  $a0,str
li  $a1,60
syscall
move  $17,$a0
addi  $sp,$sp,-8
sw  $a0,($sp)
jal  search
lw  $8,4($sp)
addi  $sp,$sp,8
addi  $sp,$sp,-20
move  $9,$sp
addi  $9,$9,2
sb  $8,1($sp)
move  $10,$17
loop2:
lb  $11,($10)
sb  $11,($9)
addi  $10,$10,1
addi  $9,$9,1
addi  $8,$8,-1
bgtz  $8,loop2
jal  Palindrome
lb  $a0,($sp)
addi  $sp,$sp,20
li  $v0,1
syscall
li  $v0,10
```

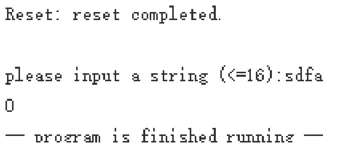
```
syscall
Palindrome:
lb  $10,1($sp)
move  $8,$sp
addi  $8,$8,2  #第一个
add  $9,$8,$10
addi  $9,$9,-1  #  最后一个
addi  $sp,$sp,-8
sw  $10,($sp)
li  $10,1
sw  $10,4($sp)
loop3:
lb  $11,($8)
lb  $12,($9)
bne  $11,$12,case1
addi  $9,$9,-1
addi  $8,$8,1
blt  $9,$8,end3
b  loop3
case1:#  不相等
sw  $0,4($sp)
end3:
lw  $8,4($sp)
addi  $sp,$sp,8
sb  $8,($sp)
jr  $ra

search:
li  $8,0
li  $10,0x0a
lw  $11,($sp)
loop1:
lb  $9,($11)
beqz  $9,end1
beq  $9,$10,end1
addi  $8,$8,1
addi  $11,$11,1
b  loop1
end1:
sw  $8,4($sp)
jr  $ra
```

结果：判断字符串是否为回文，是输出 1，不是输出 0

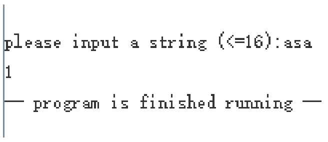


```
please input a string (<=16):abcba
1
— program is finished running —
```



```
Reset: reset completed.

please input a string (<=16):sdfa
0
— program is finished running —
```



```
please input a string (<=16):asa
1
— program is finished running —
```

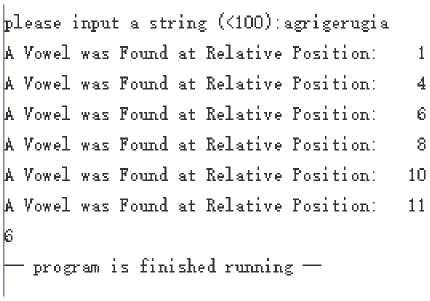
6.14

```
.data
sp:  .asciiz  "  "
enter:  .asciiz  "\n"
tip:  .asciiz  "please  input  a  string  (<100):"
mes:  .asciiz  "A  Vowel  was  Found  at  Relative  Position:       "
vowels:  .asciiz  "aeiou"
str:  .space  100
.globl  main
.text
main:
#  输入
li  $v0,4
la  $a0,tip
syscall
li  $v0,8
la  $a0,str
li  $a1,100
syscall
#  end
addi  $sp,$sp,-12
sw  $a0,($sp)
sw  $ra,8($sp)
jal  Scan
li  $v0,1
lw  $a0,4($sp)
syscall
addi  $sp,$sp,12
li  $v0,10
syscall
```

```
Scan:
lw  $9,($sp)
addi,$sp,$sp,-16
li  $17,0  #  n
li  $12,0  #  i
loop1:
la  $8,vowels
addi,$12,$12,1
lb  $10,($9)  #  str[i]
beqz  $10,end1
loop2:
lb  $11,($8)  #  yy[j]
beqz  $11,end2
beq  $10,$11,case1
addi  $8,$8,1
b  loop2
case1:
addi  $17,$17,1
#  嵌套调用打印函数
sw  $8,($sp)
sw  $9,4($sp)
sw  $17,8($sp)
sw  $12,12($sp)
addi,$sp,$sp,-12
sw  $12,($sp)
sw  $ra,8($sp)
jal  printDecimal
lw  $ra,8($sp)
addi,$sp,$sp,12
lw  $8,($sp)
lw  $9,4($sp)
lw  $17,8($sp)
lw  $12,12($sp)
#  end
end2:
addi  $9,$9,1
b  loop1
end1:
addi,$sp,$sp,16
sw  $17,4($sp)
jr  $ra
printDecimal:#可重入函数    靠右对齐
la  $a0,mes
li  $v0,4
```

```
syscall
lw  $10,($sp)
li  $9,10
sb  $0,6($sp)
div  $10,$9
mflo  $8
mfhi  $10
bnez  $8,if
li  $8,0x20
b  endif
if:
addi  $8,$8,0x30
endif:
addi  $10,$10,0x30
sb  $10,5($sp)
sb  $8,4($sp)
li  $v0,4
move  $a0,$sp
addi  $a0,$a0,4
syscall
la  $a0,enter
li  $v0,4
syscall
jr  $ra
```

结果:



```
please input a string (<100):agrigerugia
A Vowel was Found at Relative Position:    1
A Vowel was Found at Relative Position:    4
A Vowel was Found at Relative Position:    6
A Vowel was Found at Relative Position:    8
A Vowel was Found at Relative Position:   10
A Vowel was Found at Relative Position:   11
6
— program is finished running —
```

Scan 是可重入函数，未使用全局数据段分配的内存变量，局部变量在堆栈上动态分配空间。