מבנה מחשבים ערב

פרויקט 1

יעקב חיים איינאו 316360403

בן אפרת 319001319

שאלה 1)

#Q1

#Store random numbers in range of [-50,50] in an array.

.data 0x10010000 #block address

Array: .space 80 # init Array[80 bytes]==Array[20 words].

.text

li $t0,20 #$t0=20

la $t1,Array #load address of Array to $t1.

L1:

beq $t0,$0,END #if $t0==0 then jump to the END.

li $v0,42 #call for rand function.

li $a1,100 #sets $a0 to be in the range of 0<=$a0<=100.

syscall #$a0=Rand in the given range (0-100).

addi $a0,$a0,-50 #$a0-=50 to change the range -50<=$a0<=50.

sw $a0,0($t1) #$t1=$a0, store the random number in the array

addi $t1,$t1,4 #$t1+=4, go to next index in the array.

addi $t0,$t0,-1 #$t0-=1

j L1 #jump to L1 label

END:

li $v0,10 #terminate program

syscall #execute

שאלה 2)

#Q2

#Store random numbers in range of [-50,50] in an array. Now as a function

#The function gets the following arguments:

#$a0=Start of Block Address

#$a1=Number of words

#The caller needs to assign space

.data 0x10010000

.text

Data\_Block:

#initial function commands:

addi $sp,$sp,-4

sw $s0, 0($sp)

#start of funciton:

add $t0,$a0,$0 #$t0=$a0, means that $t0 has the start of the block address

add $t1,$a1,$0 #$t1=$a1, means that $t1 has the number of words.

L1:

beq $t1,$0,END\_DataBlock #if $t1==0 then jump to the END.

li $v0,42 #call for rand function.

li $a1,100 #sets $a0 to be in the range of 0<=$a0<=100.

syscall #$a0=Rand in the given range (0-100).

addi $a0,$a0,-50 #$a0-=50 to change the range -50<=$a0<=50.

sw $a0,0($t0) #$t0=$a0, store the random number in the array

addi $t0,$t0,4 #$t0+=4, go to next index in the array.

addi $t1,$t1,-1 #$t1-=1

j L1 #jump to L1 label

END\_DataBlock: #end of function

lw $s0, 0($sp)

addi $sp, $sp, 4

jr $ra #return to main

שאלה 3)

#q3 copy 20 words from one data block to another

.data 0x10010000

Array: .space 80 #init Array[80 bytes]==Array[20 words]

.text

la $a0,Array #$a0=Address of Array

li $a1,20 #$a1=20 words

jal Data\_Block #init Array, using $a0 and $a1.

la $t0,Array #load address of Res to $t10

addi $t1,$0,20 #$t1=20. Used as index for loop.

Loop:

beq $t1,$0,END #if $t1==0 then finish, else continue

lw $t4,0($t0) #$t4=$t0[i] == $t4=Array[i]

sw $t4,0x100($t0) #assign $t4 to 0x10010100+4\*index

addi $t0,$t0,4 #move one index in $t0

addi $t1,$t1,-1 #idx--

j Loop

END:

li $v0,10 #terminate program

syscall #execute

Data\_Block:

#initial function commands:

addi $sp,$sp,-4

sw $s0, 0($sp)

#start of funciton:

add $t0,$a0,$0 #$t0=$a0, means that $t0 has the start of the block address

add $t1,$a1,$0 #$t1=$a1, means that $t1 has the number of words.

L1:

beq $t1,$0,END\_DataBlock #if $t1==0 then jump to the END.

li $v0,42 #call for rand function.

li $a1,100 #sets $a0 to be in the range of 0<=$a0<=100.

syscall #$a0=Rand in the given range (0-100).

addi $a0,$a0,-50 #$a0-=50 to change the range -50<=$a0<=50.

sw $a0,0($t0) #$t0=$a0, store the random number in the array

addi $t0,$t0,4 #$t0+=4, go to next index in the array.

addi $t1,$t1,-1 #$t1-=1

j L1 #jump to L1 label

END\_DataBlock: #end of function

lw $s0, 0($sp)

addi $sp, $sp, 4

jr $ra #return to main

שאלה 4)

#Q4: Swap 2 arrays.

.data 0x10010000

Array: .space 80 #init Array[80 bytes]==Array[20 words]

.text

#init 1st array:

la $a0,Array #$a0=Address of Array

li $a1,20 #$a1=20 words

jal Data\_Block #init Array, using $a0 and $a1.

#Init 2nd array:

la $a0,Array #$a0=0x10010000

addi $a0,$a0,0x100 #$a0=0x10010100

li $a1,20 #$a1=20 words

jal Data\_Block #init Array, using $a0 and $a1

la $t0,Array #$t0=0x10010000

la $t1,Array #$t1=0x10010000

addi $t1,$t1,0x100 #$t1=0x10010100

addi $t4,$0,20 #$t4=20 used as idx

loop:

beq $t4,$0,end

lw $t2,0($t0) #$t2=$t0[idx]

lw $t3,0($t1) #$t3=$t1[idx]

sw $t2,0($t1) #t1[idx]=$t2

sw $t3,0($t0) #t0[idx]=$t3

addi $t0,$t0,4 #idx++

addi $t1,$t1,4 #idx++

addi $t4,$t4,-1 #$t4--

j loop

end:

li $v0,10 #terminate program

syscall #execute

Data\_Block:

#initial function commands:

addi $sp,$sp,-4

sw $s0, 0($sp)

#start of funciton:

add $t0,$a0,$0 #$t0=$a0, means that $t0 has the start of the block address

add $t1,$a1,$0 #$t1=$a1, means that $t1 has the number of words.

L1:

beq $t1,$0,END\_DataBlock #if $t1==0 then jump to the END.

li $v0,42 #call for rand function.

li $a1,100 #sets $a0 to be in the range of 0<=$a0<=100.

syscall #$a0=Rand in the given range (0-100).

addi $a0,$a0,-50 #$a0-=50 to change the range -50<=$a0<=50.

sw $a0,0($t0) #$t0=$a0, store the random number in the array

addi $t0,$t0,4 #$t0+=4, go to next index in the array.

addi $t1,$t1,-1 #$t1-=1

j L1 #jump to L1 label

END\_DataBlock: #end of function

lw $s0, 0($sp)

addi $sp, $sp, 4

jr $ra #return to main

שאלה 5)

#Q5: Find maximum word in array.

.data 0x10010000

Array: .space 80 #init Array[80 bytes]==Array[20 words]

.text

la $a0,Array #$a0=Address of Array

li $a1,20 #$a1=20 words

jal Data\_Block

la $t0,Array

addi $t1,$0,-60 #$t1=-60 <-50 out of range. MAX

addi $t2,$0,20 #$t2=20, idx

loop:

beq $t2,$0,end #if $t2==0 then jump to end

lw $t3,0($t0) #$t3=$t0[idx].

slt $t4,$t3,$t1 #if $t3<$t1 then $t4=1, else $t4=0

beq $t4,1,Update #if $t4==1, no new max, jump to update the index

move $t1,$t3 #$t1=$t3

Update:

addi $t0,$t0,4 #$t0[idx++]

addi $t2,$t2,-1 #$t2--

j loop

end:

li $v0,1

move $a0,$t1 #print result

syscall

li $v0,10 #terminate program

syscall

Data\_Block:

#initial function commands:

addi $sp,$sp,-4

sw $s0, 0($sp)

#start of funciton:

add $t0,$a0,$0 #$t0=$a0, means that $t0 has the start of the block address

add $t1,$a1,$0 #$t1=$a1, means that $t1 has the number of words.

L1:

beq $t1,$0,END\_DataBlock #if $t1==0 then jump to the END.

li $v0,42 #call for rand function.

li $a1,100 #sets $a0 to be in the range of 0<=$a0<=100.

syscall #$a0=Rand in the given range (0-100).

addi $a0,$a0,-50 #$a0-=50 to change the range -50<=$a0<=50.

sw $a0,0($t0) #$t0=$a0, store the random number in the array

addi $t0,$t0,4 #$t0+=4, go to next index in the array.

addi $t1,$t1,-1 #$t1-=1

j L1 #jump to L1 label

END\_DataBlock: #end of function

lw $s0, 0($sp)

addi $sp, $sp, 4

jr $ra #return to main

שאלה 6)

#Q6: Find minimum word in array.

.data 0x10010000

Array: .space 80 #init Array[80 bytes]==Array[20 words]

.text

la $a0,Array #$a0=Address of Array

li $a1,20 #$a1=20 words

jal Data\_Block

la $t0,Array

addi $t1,$0,60 #$t1=60 >50 out of range. MAX

addi $t2,$0,20 #$t2=20, idx

loop:

beq $t2,$0,end #if $t2==0 then jump to end

lw $t3,0($t0) #$t3=$t0[idx].

slt $t4,$t1,$t3 #if $t3<$t1 then $t4=1, else $t4=0

beq $t4,1,Update #if $t4==1, no new min, jump to update the index

move $t1,$t3 #$t1=$t3

Update:

addi $t0,$t0,4 #$t0[idx++]

addi $t2,$t2,-1 #$t2--

j loop

end:

li $v0,1 #print result

move $a0,$t1

syscall

li $v0,10 #terminate program

syscall

Data\_Block:

#initial function commands:

addi $sp,$sp,-4

sw $s0, 0($sp)

#start of funciton:

add $t0,$a0,$0 #$t0=$a0, means that $t0 has the start of the block address

add $t1,$a1,$0 #$t1=$a1, means that $t1 has the number of words.

L1:

beq $t1,$0,END\_DataBlock #if $t1==0 then jump to the END.

li $v0,42 #call for rand function.

li $a1,100 #sets $a0 to be in the range of 0<=$a0<=100.

syscall #$a0=Rand in the given range (0-100).

addi $a0,$a0,-50 #$a0-=50 to change the range -50<=$a0<=50.

sw $a0,0($t0) #$t0=$a0, store the random number in the array

addi $t0,$t0,4 #$t0+=4, go to next index in the array.

addi $t1,$t1,-1 #$t1-=1

j L1 #jump to L1 label

END\_DataBlock: #end of function

lw $s0, 0($sp)

addi $sp, $sp, 4

jr $ra #return to main

שאלה 7)

#Q7: Swap odd and even words.

.data 0x10010000

Array: .space 80 #init Array[80 bytes]==Array[20 words]

.text

la $a0,Array #$a0=Address of Array

li $a1,20 #$a1=20 words

jal Data\_Block #init data block

la $t0,Array #$t0=0x10010000

addi $t4,$0,10 #idx=10

Loop:

beq $t4,$0,END #if $t4==0 jump to END

lw $t1,0($t0) #$t1=$t0[even index]

lw $t2,4($t0) #$t2=$t0[odd index]

sw $t2,0($t0) #$t0[even idx]=$t2

sw $t1,4($t0) #$t0[odd idx]=$t1

addi $t0,$t0,8 #$t0+=2 moving to the next one

addi $t4,$t4,-1 #$t4--

j Loop

END:

li $v0,10 #terminate program

syscall

Data\_Block:

#initial function commands:

addi $sp,$sp,-4

sw $s0, 0($sp)

#start of funciton:

add $t0,$a0,$0 #$t0=$a0, means that $t0 has the start of the block address

add $t1,$a1,$0 #$t1=$a1, means that $t1 has the number of words.

L1:

beq $t1,$0,END\_DataBlock #if $t1==0 then jump to the END.

li $v0,42 #call for rand function.

li $a1,100 #sets $a0 to be in the range of 0<=$a0<=100.

syscall #$a0=Rand in the given range (0-100).

addi $a0,$a0,-50 #$a0-=50 to change the range -50<=$a0<=50.

sw $a0,0($t0) #$t0=$a0, store the random number in the array

addi $t0,$t0,4 #$t0+=4, go to next index in the array.

addi $t1,$t1,-1 #$t1-=1

j L1 #jump to L1 label

END\_DataBlock: #end of function

lw $s0, 0($sp)

addi $sp, $sp, 4

jr $ra #return to main

שאלה 8)

#Q8: Swap half words in a given word.

.data 0x10010000

Array: .space 80 #init Array[80 bytes]==Array[20 words]

.text

la $a0,Array #$a0=Address of Array

li $a1,20 #$a1=20 words

jal Data\_Block #init data block

la $t0,Array #$t0=0x10010000

addi $t4,$0,20 #idx=20

Loop:

beq $t4,$0,END #if $t4==0 jump to END

lh $t1,0($t0) #$t1=$t0[bottom half index]

lh $t2,2($t0) #$t2=$t0[upper half index]

sh $t2,0($t0) #$t0[upper half idx]=$t2

sh $t1,2($t0) #$t0[bottom half idx]=$t1

addi $t0,$t0,4 #$t0++ moving to the next one

addi $t4,$t4,-1 #$t4--

j Loop

END:

li $v0,10 #terminate program

syscall

Data\_Block:

#initial function commands:

addi $sp,$sp,-4

sw $s0, 0($sp)

#start of funciton:

add $t0,$a0,$0 #$t0=$a0, means that $t0 has the start of the block address

add $t1,$a1,$0 #$t1=$a1, means that $t1 has the number of words.

L1:

beq $t1,$0,END\_DataBlock #if $t1==0 then jump to the END.

li $v0,42 #call for rand function.

li $a1,100 #sets $a0 to be in the range of 0<=$a0<=100.

syscall #$a0=Rand in the given range (0-100).

addi $a0,$a0,-50 #$a0-=50 to change the range -50<=$a0<=50.

sw $a0,0($t0) #$t0=$a0, store the random number in the array

addi $t0,$t0,4 #$t0+=4, go to next index in the array.

addi $t1,$t1,-1 #$t1-=1

j L1 #jump to L1 label

END\_DataBlock: #end of function

lw $s0, 0($sp)

addi $sp, $sp, 4

jr $ra #return to main

שאלה 9)

#Q9: Sum all the words in the data block and print it.

.data 0x10010000

Array: .space 80 #init Array[80 bytes]==Array[20 words]

.text

la $a0,Array #$a0=Address of Array

li $a1,20 #$a1=20 words

jal Data\_Block #init data block

la $t0,Array #$t0=0x10010000

li $t2,20 #$t2=20 idx

move $t1,$0 #$t1=0

Loop:

beq $t2,$0,END #if $t2==0 jump to END

lw $t3,0($t0) #$t3=$t0[idx]

add $t1,$t1,$t3 #$t1+=$t3

addi $t2,$t2,-1 #$t2--

addi $t0,$t0,4 #$t0[idx++]

j Loop

END:

li $v0,1 #print result

move $a0,$t1

syscall

li $v0,10 #terminate program

syscall

Data\_Block:

#initial function commands:

addi $sp,$sp,-4

sw $s0, 0($sp)

#start of funciton:

add $t0,$a0,$0 #$t0=$a0, means that $t0 has the start of the block address

add $t1,$a1,$0 #$t1=$a1, means that $t1 has the number of words.

L1:

beq $t1,$0,END\_DataBlock #if $t1==0 then jump to the END.

li $v0,42 #call for rand function.

li $a1,100 #sets $a0 to be in the range of 0<=$a0<=100.

syscall #$a0=Rand in the given range (0-100).

addi $a0,$a0,-50 #$a0-=50 to change the range -50<=$a0<=50.

sw $a0,0($t0) #$t0=$a0, store the random number in the array

addi $t0,$t0,4 #$t0+=4, go to next index in the array.

addi $t1,$t1,-1 #$t1-=1

j L1 #jump to L1 label

END\_DataBlock: #end of function

lw $s0, 0($sp)

addi $sp, $sp, 4

jr $ra #return to main

שאלה 10)

#Q10: Sum a[i]=a[i]+a[i+1], i=0,2,4...

.data 0x10010000

Array: .space 80 #init Array[80 bytes]==Array[20 words]

.text

la $a0,Array #$a0=Address of Array

li $a1,20 #$a1=20 words

jal Data\_Block #init data block

la $t0,Array #$t0=0x10010000

li $t2,10 #$t2=10 idx

Loop:

beq $t2,$0,END #if $t2==0 jump to END

lw $t3,0($t0) #$t3=$t0[idx]

lw $t4,4($t0) #$t4=$t0[idx+1]

add $t3,$t3,$t4 #$t3+=$t4

sw $t3,0($t0) #$t0[idx]=$t3

addi $t2,$t2,-1 #$t2--

addi $t0,$t0,8 #$t0[idx+2]

j Loop

END:

li $v0,10 #terminate program

syscall

Data\_Block:

#initial function commands:

addi $sp,$sp,-4

sw $s0, 0($sp)

#start of funciton:

add $t0,$a0,$0 #$t0=$a0, means that $t0 has the start of the block address

add $t1,$a1,$0 #$t1=$a1, means that $t1 has the number of words.

L1:

beq $t1,$0,END\_DataBlock #if $t1==0 then jump to the END.

li $v0,42 #call for rand function.

li $a1,100 #sets $a0 to be in the range of 0<=$a0<=100.

syscall #$a0=Rand in the given range (0-100).

addi $a0,$a0,-50 #$a0-=50 to change the range -50<=$a0<=50.

sw $a0,0($t0) #$t0=$a0, store the random number in the array

addi $t0,$t0,4 #$t0+=4, go to next index in the array.

addi $t1,$t1,-1 #$t1-=1

j L1 #jump to L1 label

END\_DataBlock: #end of function

lw $s0, 0($sp)

addi $sp, $sp, 4

jr $ra #return to main

שאלה 11)

#Q11: Count all the positive numbers

.data 0x10010000

Array: .space 80 #init Array[80 bytes]==Array[20 words]

.text

la $a0,Array #$a0=Address of Array

li $a1,20 #$a1=20 words

jal Data\_Block #init data block

la $t0,Array #$t0=0x10010000

li $t2,20 #$t2=20 idx

li $t1,0 #t1=0 counter

Loop:

beq $t2,$0,END #if $t2==0 jump to END

lw $t3,0($t0) #$t3=$t0[idx]

slt $t4,$0,$t3 # if 0<$t3 then $t4

beq $t4,$0,Update #if $t4==0 then jump to update, else increase the count

addi $t1,$t1,1 #$t1++ increase counter

Update:

addi $t2,$t2,-1 #$t2--

addi $t0,$t0,4 #$t0[idx++]

j Loop

END:

li $v0,1 #print result

move $a0,$t1

syscall

li $v0,10 #terminate program

syscall

Data\_Block:

#initial function commands:

addi $sp,$sp,-4

sw $s0, 0($sp)

#start of funciton:

add $t0,$a0,$0 #$t0=$a0, means that $t0 has the start of the block address

add $t1,$a1,$0 #$t1=$a1, means that $t1 has the number of words.

L1:

beq $t1,$0,END\_DataBlock #if $t1==0 then jump to the END.

li $v0,42 #call for rand function.

li $a1,100 #sets $a0 to be in the range of 0<=$a0<=100.

syscall #$a0=Rand in the given range (0-100).

addi $a0,$a0,-50 #$a0-=50 to change the range -50<=$a0<=50.

sw $a0,0($t0) #$t0=$a0, store the random number in the array

addi $t0,$t0,4 #$t0+=4, go to next index in the array.

addi $t1,$t1,-1 #$t1-=1

j L1 #jump to L1 label

END\_DataBlock: #end of function

lw $s0, 0($sp)

addi $sp, $sp, 4

jr $ra #return to main

שאלה 12)

#Q12: Multiply all the data by 2.

.data 0x10010000

Array: .space 80 #init Array[80 bytes]==Array[20 words]

.text

la $a0,Array #$a0=Address of Array

li $a1,20 #$a1=20 words

jal Data\_Block #init data block

la $t0,Array #$t0=0x10010000

li $t2,20 #$t2=20 idx

li $t1,2 #$t1=2

Loop:

beq $t2,$0,END #if $t2==0 jump to END

lw $t3,0($t0) #$t3=$t0[idx]

mul $t3,$t3,$t1 #$t3\*=2

sw $t3,0($t0) #$t0[idx]=$t3

addi $t2,$t2,-1 #$t2--

addi $t0,$t0,4 #$t0[idx++]

j Loop

END:

li $v0,10 #terminate program

syscall

Data\_Block:

#initial function commands:

addi $sp,$sp,-4

sw $s0, 0($sp)

#start of funciton:

add $t0,$a0,$0 #$t0=$a0, means that $t0 has the start of the block address

add $t1,$a1,$0 #$t1=$a1, means that $t1 has the number of words.

L1:

beq $t1,$0,END\_DataBlock #if $t1==0 then jump to the END.

li $v0,42 #call for rand function.

li $a1,100 #sets $a0 to be in the range of 0<=$a0<=100.

syscall #$a0=Rand in the given range (0-100).

addi $a0,$a0,-50 #$a0-=50 to change the range -50<=$a0<=50.

sw $a0,0($t0) #$t0=$a0, store the random number in the array

addi $t0,$t0,4 #$t0+=4, go to next index in the array.

addi $t1,$t1,-1 #$t1-=1

j L1 #jump to L1 label

END\_DataBlock: #end of function

lw $s0, 0($sp)

addi $sp, $sp, 4

jr $ra #return to main

שאלה 13)

#Q13: Add 0x1000 to every word in the data block.

.data 0x10010000

Array: .space 80 #init Array[80 bytes]==Array[20 words]

.text

la $a0,Array #$a0=Address of Array

li $a1,20 #$a1=20 words

jal Data\_Block #init data block

la $t0,Array #$t0=0x10010000

li $t2,20 #$t2=20 idx

Loop:

beq $t2,$0,END #if $t2==0 jump to END

lw $t3,0($t0) #$t3=$t0[idx]

addi $t3,$t3,0x1000 #$t3+=0x1000

sw $t3,0($t0) #$t0[idx]=$t3

addi $t2,$t2,-1 #$t2--

addi $t0,$t0,4 #$t0[idx++]

j Loop

END:

li $v0,10 #terminate program

syscall

Data\_Block:

#initial function commands:

addi $sp,$sp,-4

sw $s0, 0($sp)

#start of funciton:

add $t0,$a0,$0 #$t0=$a0, means that $t0 has the start of the block address

add $t1,$a1,$0 #$t1=$a1, means that $t1 has the number of words.

L1:

beq $t1,$0,END\_DataBlock #if $t1==0 then jump to the END.

li $v0,42 #call for rand function.

li $a1,100 #sets $a0 to be in the range of 0<=$a0<=100.

syscall #$a0=Rand in the given range (0-100).

addi $a0,$a0,-50 #$a0-=50 to change the range -50<=$a0<=50.

sw $a0,0($t0) #$t0=$a0, store the random number in the array

addi $t0,$t0,4 #$t0+=4, go to next index in the array.

addi $t1,$t1,-1 #$t1-=1

j L1 #jump to L1 label

END\_DataBlock: #end of function

lw $s0, 0($sp)

addi $sp, $sp, 4

jr $ra #return to main

שאלה 14)

#Q14: Get user input for a number, and count how many times that number is in the data block, and print the result.

#Note: The string is saved in the memory, check the ascii view to see it.

.data 0x10010000

Array: .space 80 #init Array[80 bytes]==Array[20 words]

output: .asciiz "Please Enter a number to find at the array: "

.text

la $a0,Array #$a0=Address of Array

li $a1,20 #$a1=20 words

jal Data\_Block #init data block

li $v0,4 #set up print

la $a0, output

syscall #print the string

li $v0,5 #call for input from user

syscall #input call

la $t0,Array #$t0=0x10010000

add $t1,$0,$0 #$t1=0, used as counter

li $t2,20 #$t2=20 idx

Loop:

beq $t2,$0,END #if $t2==0 jump to END

lw $t3,0($t0) #$t3=$t0[idx]

bne $t3,$v0,Update #if $t3!=$v0 then update index, else update counter

addi $t1,$t1,1 #$t1++ increase counter

Update:

addi $t2,$t2,-1 #$t2--

addi $t0,$t0,4 #$t0[idx++]

j Loop

END:

li $v0,1 #print result

move $a0,$t1

syscall

li $v0,10 #terminate program

syscall

Data\_Block:

#initial function commands:

addi $sp,$sp,-4

sw $s0, 0($sp)

#start of funciton:

add $t0,$a0,$0 #$t0=$a0, means that $t0 has the start of the block address

add $t1,$a1,$0 #$t1=$a1, means that $t1 has the number of words.

L1:

beq $t1,$0,END\_DataBlock #if $t1==0 then jump to the END.

li $v0,42 #call for rand function.

li $a1,100 #sets $a0 to be in the range of 0<=$a0<=100.

syscall #$a0=Rand in the given range (0-100).

addi $a0,$a0,-50 #$a0-=50 to change the range -50<=$a0<=50.

sw $a0,0($t0) #$t0=$a0, store the random number in the array

addi $t0,$t0,4 #$t0+=4, go to next index in the array.

addi $t1,$t1,-1 #$t1-=1

j L1 #jump to L1 label

END\_DataBlock: #end of function

lw $s0, 0($sp)

addi $sp, $sp, 4

jr $ra #return to main

שאלה 15)

#Q14: Get user input for a number, and count how many times that number is in the data block, and print the result.

#Note: The string is saved in the memory, check the ascii view to see it.

.data 0x10010000

outputA: .asciiz "Enter two numbers and I'll show you the sum, difference, product, quotient, and remainder.\n\nPlease Enter a number A: "

outputB: .asciiz "Please Enter a number B: "

outputSum: .asciiz "A+B= "

outputDif: .asciiz "\nA-B= "

outputProduct: .asciiz "\nA\*B= "

outputDiv: .asciiz "\nA/B= "

outputRem: .asciiz "\nA%B= "

invalidDiv: .asciiz "\nCan't divided by 0!"

.text

li $v0,4 #set up print

la $a0, outputA

syscall #print the string

li $v0,5 #call for input from user

syscall #input call A

add $t2,$0,$v0 #$t2=$v0

li $v0,4 #set up print

la $a0, outputB

syscall #print the string

li $v0,5 #call for input from user

syscall #input call B

add $t3,$0,$v0 #$t3=$v0

#Sum:

add $t1,$t2,$t3 #$t1=$t2+$t3 is Sum=A+B

li $v0,4 #set up print

la $a0, outputSum

syscall #print the string

li $v0,1 #print result

move $a0,$t1

syscall

#Sub:

sub $t1,$t2,$t3 #$t1=$t2-$t3 is Dif=A-B

li $v0,4 #set up print

la $a0, outputDif

syscall #print the string

li $v0,1 #print result

move $a0,$t1

syscall

#Product:

mul $t1,$t2,$t3 #$t1=$t2\*$t3 is Product=A\*B

li $v0,4 #set up print

la $a0, outputProduct

syscall #print the string

li $v0,1 #print result

move $a0,$t1

syscall

bne $t3,$0,Valid #if $t3!=0 then the input is valid, can dvide, else print error.

li $v0,4 #set up print

la $a0, invalidDiv

syscall #print the string

j END

Valid:

#Quotient:

li $v0,4 #set up print

la $a0, outputDiv

syscall #print the string

div $t2,$t3 #$t1=$t2/$t3 is Quotient=A/B

mflo $a0 #get quotient

mfhi $t1 #get remainder

li $v0,1 #print result

syscall

#Remainder:

li $v0,4 #set up print

la $a0, outputRem

syscall #print the string

li $v0,1 #print result

move $a0,$t1

syscall

END:

li $v0,10 #terminate program

syscall

שאלה 16)

#Q16: Turn all the characters in 'Allice.txt' to caps. Please read the note!!!!!

###NOTE: the file must be at the same directory as MIPS.jar###

.data 0x10010000

Allice: .asciiz "Allice.txt"

AlliceU: .asciiz "AlliceU.txt"

ErrorOutput: .asciiz "Error! Exiting the program..."

ReserveSpace: .space 2000 #reserved space

.text

#Open 'Allice.txt'

la $a0,Allice #get address of Allice

li $v0,13 #Open file mode

li $a1,0 #Open for read only

li $a2,0 #mode is ignored (from Help)

syscall

#Check for errors:

slt $t0,$0,$v0 #check for error, if $v0 is negative then error.

bne $t0,$0,Continue1 #if $t0!=0 then it worked, continue.

li $v0,4 #set up print

la $a0, ErrorOutput

syscall #print the string

j END

Continue1:

#Read 'Allice.txt'

move $a0,$v0 #save the file descriptor

###Debug###

#li $v0,16 #close 'Allice.txt'

#syscall

li $v0,14 #Read file mode

la $a1,ReserveSpace #get address for output buffer

li $a2,2000 #maximum characters to read

syscall

#Check for error

slt $t0,$0,$v0 #check for error, if $v0 is negative then error.

bne $t0,$0,Continue2 #if $t0!=0 then it worked, continue.

li $v0,4 #set up print

la $a0, ErrorOutput

syscall #print the string

j END

#Replace all the minor characters

Continue2:

move $s5,$v0 #save the file descriptor

addi $t8,$0,97 #$t8='a'

addi $t9,$0,122 #$t9='z'

Loop:

lb $t2,0($a1) #$t2=$a1[idx]

blt $t2,$t8,Cap #if $t2<'a' then jump to Cap, else continue

bgt $t2,$t9,Cap #if $t2>'z' then jump to Cap, else continue

#minor characters is found

addi $t2,$t2,-32 #Turn $t2 to a Cap.

sb $t2,0($a1) #$a1[i]=$t2 now Cap.

Cap:

addi $a1,$a1,1 #a1[idx++]

lb $t1,0($a1) #$t1=$a1[idx]

bne $t1,$0,Loop #if $t1!='\0' then continue the loop

#Write 'AlliceU.txt'

la $a0, AlliceU #address of 'AlliceU.txt'

li $v0,13 #open file

li $a1,1 #Create file if needed

li $a2,0

syscall

move $a0,$v0 #$a0=$t1 for file descriptor for AlliceU

la $a1,ReserveSpace #load buffer address for $a1

move $a2,$s5 #load num of characters to write to $a2

li $v0,15 #read file

syscall

END:

li $v0,16 #close 'AlliceU.txt'

syscall

li $v0,16 #close 'Allice.txt'

la $a0,Allice

syscall

li $v0,10 #terminate program

syscall