**CSE331 Computer Organization**

Assignment 1 /REPORT

***Author :lemye ceren gümüş -151044071***

* **Data kısmında arr, subStArr, sameArr, diffArr 160 size’lık yer alınır ve File path oluşturulur.**

.data

fin: .asciiz "C:\\Users\\user\\Downloads\\input.txt" # Filename path

buffer: .space 128

buffer1: .asciiz "\n"

val : .space 128

newline: .asciiz "\n"

space : .asciiz " "

arr: .space 160

subStArr: .space 160

sameArr: .space 160

diffArr: .space 160

* **Verilen path dizinindeki dosya açılır**

readFile:

li $v0, 13 # system call for open file

la $a0, fin # board file name

li $a1, 0 # Open for reading

li $a2, 0

syscall # open a file (file descriptor returned in $v0)

move $s6, $v0 # save the file descriptor

* **Dosya okunarak buffer dizisine doldurulur ve ekranda print edilir.**

#read from file

li $v0, 14 # system call for read from file

move $a0, $s6 # file descriptor

la $a1, buffer # address of buffer to which to read

li $a2, 1024 # hardcoded buffer length

syscall # read from file

# Print string from file

li $v0, 4 # system call for print string

la $a0, buffer # string

syscall # print string

addi $t2,$zero,0

* **Set ve subset kümeleri buffer array parse edilerek diziler doldurulur.**

L1:

lb $t0, buffer($t2) #loading value

add $t2, $t2, 1

beq $t0,32,L1

beq $t0,34,L2

sw $t0,0($s1)

addi $s1,$s1,4

addi $t5,$t5,1

bne $t0, $zero, L1

la $s2,subStArr

addi $t3,$zero,0

j L2

L2: lb $t0, buffer($t2) #loading value

add $t2, $t2, 1

beq $t0,32,L2

beq $t0,10,L2

sw $t0,0($s2)

addi $s2,$s2,4

addi $t3,$t3,1

bne $t0, $zero, L2

addi $t0,$zero,42

sw $t0,0($s2)

addi $s2,$s2,4

move $t3,$s3 # store set size

addi $t2,$zero,0

* **İki küme arasında benzer olanları bularak yeni bir sameArr oluşturur**.

Intersection:

#beq $t3,0,printSubSet

jal checkSub

j Exit1

checkSub:

addi $t9,$zero,0 # count

addi $t2,$zero,0 # index subset

addi $t7,$zero,0 # index set

loop2:

lw $t4,subStArr($t2)

loop:

beq $t4,34,out

lw $t6,arr($t7)

bne $t4,$t6 in

sw $t4,sameArr($t2)

la $a0, 0($t4)

li $v0, 11 # system call code for print\_in

syscall # p

addi $t9,$t9,1

addi $t2,$t2,4

j loop2

in:

addi $t7,$t7,4

j loop

out:

jr $ra

Arr ve subStrArr label ile iki tane dizi oluşturulur. File’dan tüm datalar okunur. File okunurken her satırın sonunda “ işareti ile kontrol sağlanır. Set ve subset kümelerini oluşturmak için ayrı ayrı label oluşturulmuştur (L1 ve L2). Arr ve subStrArr içerisine datalar doldurulur. t5 ve t3 register’larına eleman sayıları tutulur. Intersection label içerisinde checkSub fonksiyonu çağrılır. Algoritma bu kısımda oluşturulur.

**Differs label için: iki set arası farkı bulur,setler basta “ isareti ile ayrıldıgından her setin farkını önceden yer ayrılmıs temp arrayinde tutar,setler arası farkı en büyük olan belirlenerek minset için \* işareti ile işaretlenir**

differs :

#beq $t3,0,printSubSet

jal checkSubz

j Exit9

checkSubz:

addi $t2,$zero,0 # index subset

addi $t7,$zero,0 # index set

addi $t8,$zero,0 # index set

loop2z:

addi $t7,$zero,0

lw $t4,arr($t2)

addi $t2,$t2,4

beq $t4,42,outz

loopz:

lw $t6,sameArr($t7)

addi $t7,$t7,4

beq $t4,$t6 loop2z

beq $t4,42,loop2z

bne $t6,42,loopz

sw $t4,diffArr($t8)

addi $t8,$t8,4

j loop2z

outz:

jr $ra

Exit9:

addi $t4,$zero,42

sw $t4,diffArr($t8)

addi $t8,$t8,4

Algoritma

Set kümesi ile sub set kümeleri karşılaştırılır. Sub set kümeler set kümesinin içerisinde olmayan yeni bir set kümesi oluşturulur. Set’in size 0 olana kadar devam eder.

File

4 5 6 7 9 0 1 “

4 5 “

4 6 7 “

7 “

0 ”

Set:

4 5 6 7 9 0 1

Subset

4 5

4 6 7

7

0

|  |  |  |
| --- | --- | --- |
| Set Array | Subset Array | Durum |
| 4 5 6 7 9 0 1 | 4 5” 4 6 7 ” 7” 0” | 6 7 9 0 1 |
| 6 7 9 0 1 | 4 5” 4 6 7 ” 7” 0” | 6 9 0 1 |
| 6 9 0 1 | 4 5” 4 6 7 ” 7” 0” | 6 9 1 |
| … | … | …. |

**Minset için kullanılan subsetler outputta \* isareti ile isaretlenmistir**