

CSE 331/503 Computer Organization Homework 2

REPORT

YUNUS EMRE GEYİK / 1801042635

The Part That I Could Not Do

I could not take input from the file. In fact, homework took so much time that I had to study for other lessons. If it wasn't so tight, I'd take input from file too :)

PSEUDO CODE

Begin

define array length of size n

initially set 0 to all entries of length

set 1 to array length 0. indeks

define array 'arr' of size n

initially set 0 to all entries of arr

define array 'arr2' of size n

initially set 0 to all entries of arr2

for i = 1 to n, do

for j = 0 to i, do

if subarray[j] < subarray[i] and length[j] > length[i]

arr2[counter2]=Arr[j]

length[i] = length[j]

counter2++

done

for k=0 to counter2,do

if arr[k]==0

arr[k] = arr2[k]

if arr[k]!=0

if arr[k] > arr2[k]

arr[k] = arr2[k]

done

counter2=0

length[i]++

done

lis = 0

for i = 0 to n, do

if lis<length[i]

lis = length[i]

done

arr[lis-1]=Arr[n-1]

for t=0 to lis, do

print arr[t]

print lis

return arr

End

Time Complexity : $O(n^2)$

Space Complexity : $O(1)$

Assembly Memory (Byte)

Array = 64 + 24 + 36 + 24 + 28 + 32 = 208

Space = 256 (For another array)

Array Sizes = 24

Strings = 72

Total = 560 byte

PSEUDO CODE

To

MIPS Assembly

Begin

addi \$t0,\$zero,0 (line 1278)

define array length of size n

la \$t0,length (line 1359)

initially set 0 to all entries of length

sw \$zero,length(\$t0) (line 1307)

set 1 to array length 0. indeks

addi \$t0,\$zero,1 sw \$t0,length(\$zero)

define array 'arr' of size n

la \$s7,arr (line 1446)

initially set 0 to all entries of arr

sw \$zero,arr(\$t0) (line 1295)

define array 'arr2' of size n

la \$t7,arr2 (line 1380)

initially set 0 to all entries of arr2

sw \$zero,arr2(\$t0) (line 1283)

for i = 1 to n, do

line 1328 at MIPS

for j = 0 to i, do

if subarray[j] < subarray[i] and length[j] > length[i]

arr2[counter2]=Arr[j]

length[i] = length[j]

counter2++

done

for k=0 to counter2,do

if arr[k]==0

arr[k] = arr2[k]

if arr[k]!=0

if arr[k] > arr2[k]

arr[k] = arr2[k]

done

counter2=0

length[i]++

done

line 1541 at MIPS

lis = 0

\$t1

for i = 0 to n, do

line 1551 at MIPS

if lis<length[i]

lis = length[i]

done

line 1578 at MIPS

arr[lis-1]=Arr[n-1]

line 1589 - 1607 at MIPS

for t=0 to lis, do

print arr[t]

print lis

add \$v1,\$t1,\$zero

return arr

la \$v0,arr

End

Code Explanation

Main idea of code is , fort the finding longest increasing sub sequence , take the biggest value of the sub arrays column and put it an array an create the longest increasing sub sequence and find its size. So relying on the comments contained in the code, I believe they can be understood at a certain level.

MIPS REGISTERS

\$s7 register for arr (hold the biggest value of arr2 array's column and create longest increasing sub sequence)

\$t7 register for arr2 (hold the sub sequence every cycle of loop)

\$t0 register length (using the find the longest sub sequence size)

\$a1 and \$a3 registers for the main array that is Arr1,Arr2,Arr3.....Arr6

\$a2 register for the main array size

\$t1 register is 'lis' value that is the size of the longest increasing sub sequence (using in the procedure)

\$v0 register return the longest sub sequence address

\$v1 register return the size of the longest increasing sub sequence

\$t1 register hold the address of arr3 in the main

TEST CASES

SCREEN

```
Mars Messages Run I/O
Array 1 : [ 0, 8, 4, 12, 2, 10, 6, 14, 1, 9, 5, 13, 3, 11, 7, 15 ]
Sub Sequences :
candidate sequence : [ 0 8 ] , Size = 2
candidate sequence : [ 0 4 ] , Size = 2
candidate sequence : [ 0 8 12 ] , Size = 3
candidate sequence : [ 0 2 ] , Size = 2
candidate sequence : [ 0 8 10 ] , Size = 3
candidate sequence : [ 0 4 6 ] , Size = 3
candidate sequence : [ 0 8 12 14 ] , Size = 4
candidate sequence : [ 0 1 ] , Size = 2
candidate sequence : [ 0 8 6 9 ] , Size = 4
candidate sequence : [ 0 4 5 ] , Size = 3
candidate sequence : [ 0 8 12 9 13 ] , Size = 5
candidate sequence : [ 0 2 3 ] , Size = 3
candidate sequence : [ 0 8 10 9 11 ] , Size = 5
candidate sequence : [ 0 4 6 7 ] , Size = 4
candidate sequence : [ 0 8 12 14 13 15 ] , Size = 6
```

OUTPUT FILE

```
1 size:6
2 Array: 0 2 6 9 13 15
```

SCREEN

```
Array 2 : [ 3, 10, 7, 9, 4, 11 ]
Sub Sequences :
candidate sequence : [ 3 10 ] , Size = 2
candidate sequence : [ 3 7 ] , Size = 2
candidate sequence : [ 3 7 9 ] , Size = 3
candidate sequence : [ 3 4 ] , Size = 2
candidate sequence : [ 3 10 9 11 ] , Size = 4
```

OUTPUT FILE

```
3 size:4
4 Array: 3 7 9 11
```

SCREEN

```
Array 3 : [ 10, 22, 9, 33, 21, 50, 41, 60, 80 ]
Sub Sequences :
candidate sequence : [ 10 22 ] , Size = 2
candidate sequence : [ 9 ] , Size = 1
candidate sequence : [ 10 22 33 ] , Size = 3
candidate sequence : [ 10 21 ] , Size = 2
candidate sequence : [ 10 22 33 50 ] , Size = 4
candidate sequence : [ 10 22 33 41 ] , Size = 4
candidate sequence : [ 10 22 33 50 60 ] , Size = 5
candidate sequence : [ 10 22 33 50 60 80 ] , Size = 6
```

OUTPUT FILE

```
5 size:6
6 Array: 10 22 33 50 60 80
```

SCREEN

```
Mars Messages Run I/O
Array 5 : [ 2, 3, 3, 3, 3, 3, 3 ]
Sub Sequences :
  candidate sequence : [ 2 3 ], Size = 2
  candidate sequence : [ 2 3 ], Size = 2
  candidate sequence : [ 2 3 ], Size = 2
  candidate sequence : [ 2 3 ], Size = 2
  candidate sequence : [ 2 3 ], Size = 2
  candidate sequence : [ 2 3 ], Size = 2
```

OUTPUT FILE

```
9 size:2
10 Array: 2 3
```

SCREEN

```
Mars Messages Run I/O
Array 6 : [ 10,11,2,5,3,7,58,78 ]
Sub Sequences :
  candidate sequence : [ 10 11 ], Size = 2
  candidate sequence : [ 2 ], Size = 1
  candidate sequence : [ 2 5 ], Size = 2
  candidate sequence : [ 2 3 ], Size = 2
  candidate sequence : [ 2 5 7 ], Size = 3
  candidate sequence : [ 10 11 7 58 ], Size = 4
  candidate sequence : [ 10 11 7 58 78 ], Size = 5

Clear

-- program is finished running --
```

OUTPUT FILE

```
11 size:5
12 Array: 2 5 7 58 78
13
```

ALL OUTPUT FILE

```
D:\Emre\OneDrive - GTÜ\Masaüstü\mars- orgo\testout1.txt - MARS 4.5
File Edit Run Settings Tools Help
Run speed at max (no interaction)

mips12.asm testout1.txt
1 size:6
2 Array: 0 2 6 9 13 15
3 size:4
4 Array: 3 7 9 11
5 size:6
6 Array: 10 22 33 50 60 80
7 size:4
8 Array: 0 1 2 3
9 size:2
10 Array: 2 3
11 size:5
12 Array: 2 5 7 58 78
13
14
```

Bonus Part

In the assignment file, Showing some inner results in console as an bonus score; regarding to that, as can be seen in the test cases , this bonus part is made, some inner results showing in console at my implementation.