RAPORT FILE FOR HOMEWORK 2

In this code segment, have progressed from the input file to left bracket char:

```
for_loop_to_left_bracket:
addi $s0, $s0, 1
lb $t5, 0($s0)
bne $t0, $t5, for_loop_to_left_bracket
addi $s0, $s0, 1
```

If number is two or more digit then, we can generate this number as integer with code :

```
multi_and_add:
li $t7, 10
mul $t6, $t6, $t7
addi $t7, $t5, -48
add $t6, $t6, $t7
addi $s0, $s0, 1
lb $t5, 0($s0)
beq $t5, $t2, store # if $t5 == space
beq $t5, $t3, store #if $t5 == comma
beq $t5, $t1, end_for
j multi_and_add
```

pseudocode to generate subsequences of array in increasing order

(between 104-231 in assembly code):

```
for(int i=0;i<size;i++){
    for(int a=1;a<size;a++){
        for(int j=i;k=i+a;k<size;k++){  // taking first subsequence
            if(s1[j]<s1[k]){
                store;
                j=k;
            }
        }
    }
}
```

Pseudocode to store longest subsequence of array:

(between 172-213 in assembly code):

Counter set zero for first array

Counter set zero for second array

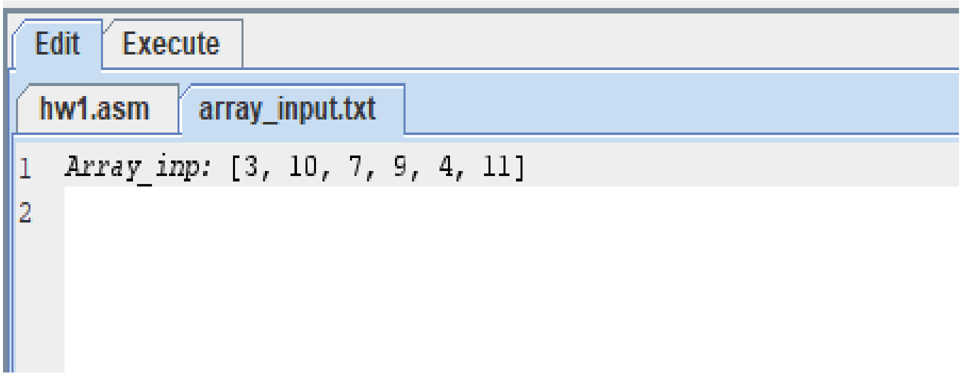
If loop is in first subsequnce then store elements to first array, and in loop increment counter one by one.

If loop is second,third.. subsequnce store elements to second array, and in loop increment counter one by one.

If first counter less than second counter then exchange elements in two array and move second array size to first array size.

and go on with other subsequence.

INPUT:



```
Array_inp: [3, 10, 7, 9, 4, 11]
```

RESULTS:

First result is subsequences of array in inreasing order :

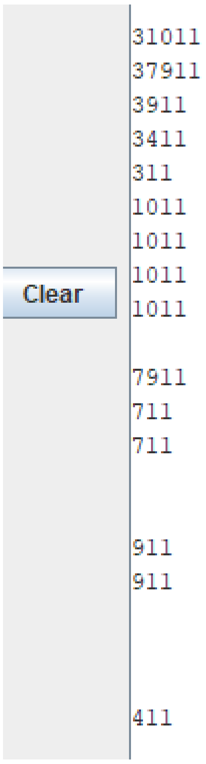
[3,10,11], [3,7,9,11], [3,9,11], [3,4,11], [3,11]

[10,11]

[7,9,11], [7,11]

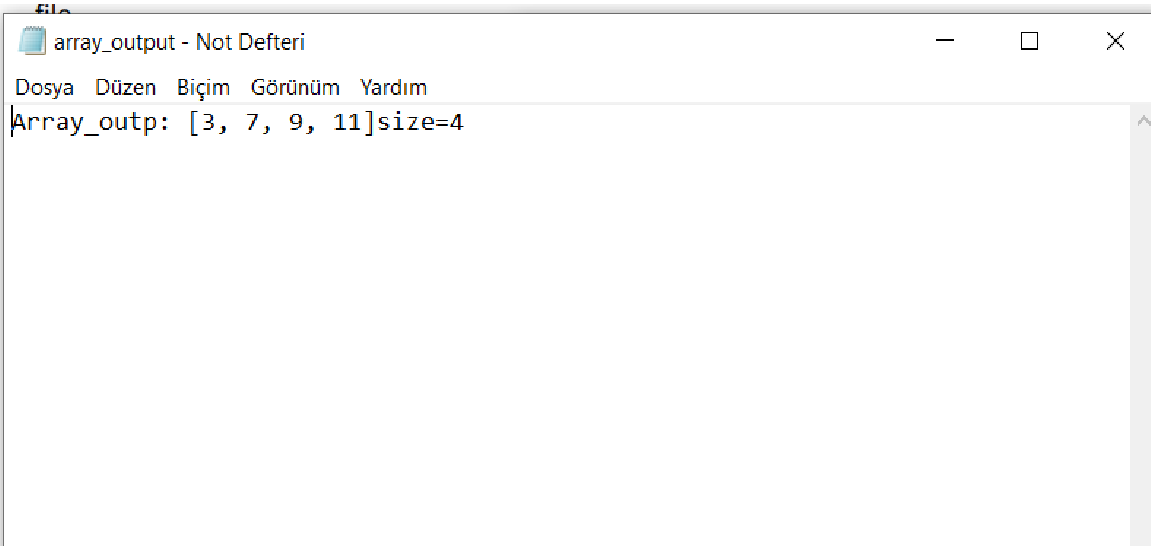
[9,11]

[4,11]



```
31011
37911
3911
3411
311
1011
1011
1011
1011

7911
711
711


911
911



411
```

file



```
Array_outp: [3, 7, 9, 11]size=4
```