CSE-331

HOMEWORK 2 REPORT

Homework also has a bonus party. The most important thing to pay attention to in my algorithm is to start scanning the array from the end. Therefore, it gives different results with algorithms starting from the beginning.

To explain the recursive function, it compares the last element of the array with our target value. If target is smaller, it ignores that element and calls the function again with a decrease of size. The other one in if works with the same or logic. If 1 turns from the first recursive arm we call, there is no need to look at the other recursive arm. But if it returns 0, this means that the value in our last element may be providing the sum. So we keep that element elsewhere. Before calling the recursive arm, we remove the last element of the array from the target and reduce it to size.

All this is written in the assembly code as a comment line next to it.

After entering the elements in the output, if possible, it prints the values that provide this first. Then possible prints.

Test Result

ASSEMBLY

```
Enter size: 10

Enter Target: 2

Enter Value
9
21
32
2
30
17
28
22
2
12
2
Possible!
-- program is finished running --
```

```
Possible!
-- program is finished running --

Enter size: 10

Enter Target: 12

Enter Value
30
30
17
31
29
30
26
30
10
25

Not Possible!
-- program is finished running --
```

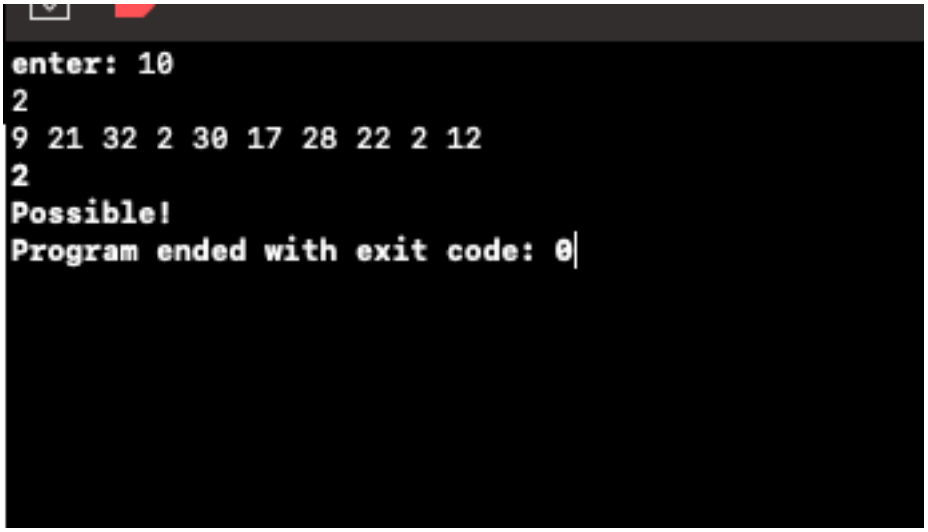
```
Possible!
-- program is finished running --

Enter size: 10

Enter Target: 22

Enter Value
19
19
9
22
29
8
31
6
18
30
22
Possible!
-- program is finished running --
```

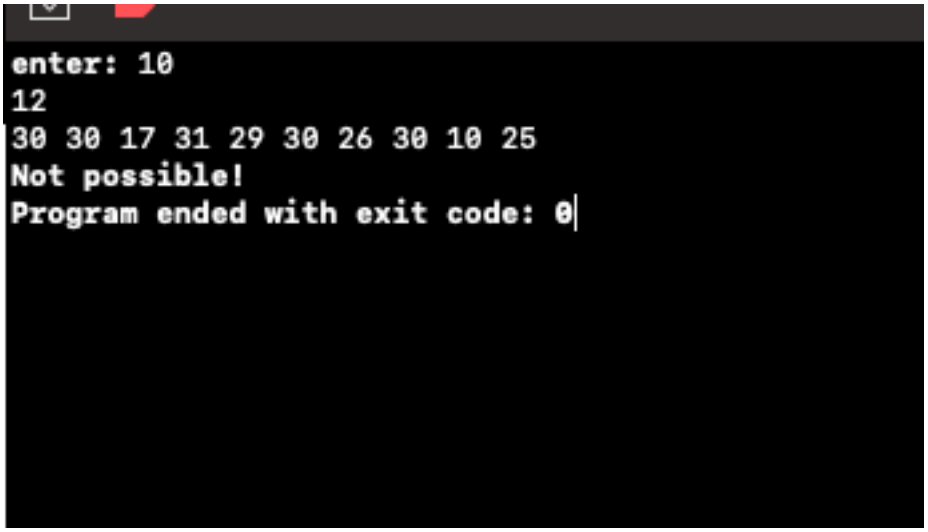
```
Possible!
-- program is finished running --

Enter size: 10

Enter Target: 32

Enter Value
3
13
21
22
22
14
27
31
8
28
3 21 8
Possible!
-- program is finished running --
```

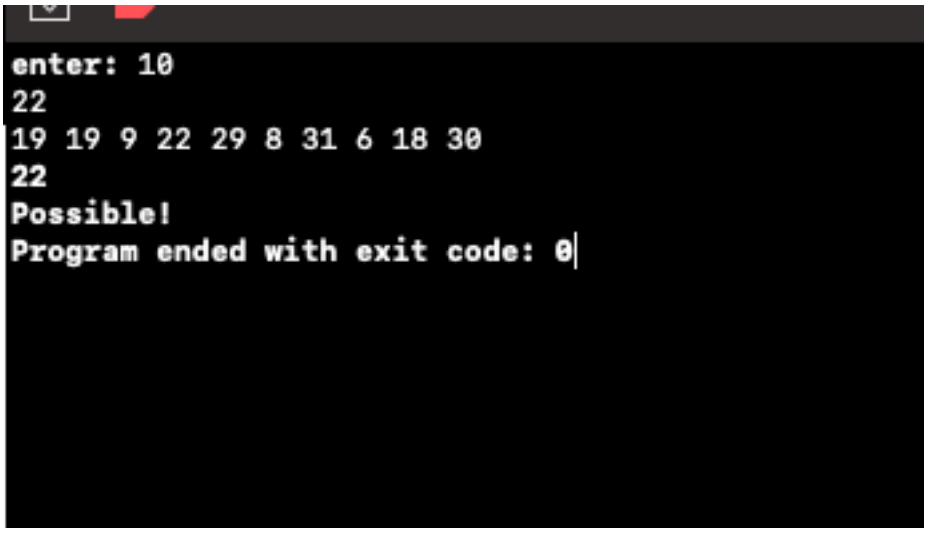
```
Possible!
-- program is finished running --

Enter size: 10

Enter Target: 42

Enter Value
27
19
6
7
19
12
28
23
6
5
19 23
Possible!
-- program is finished running --
```

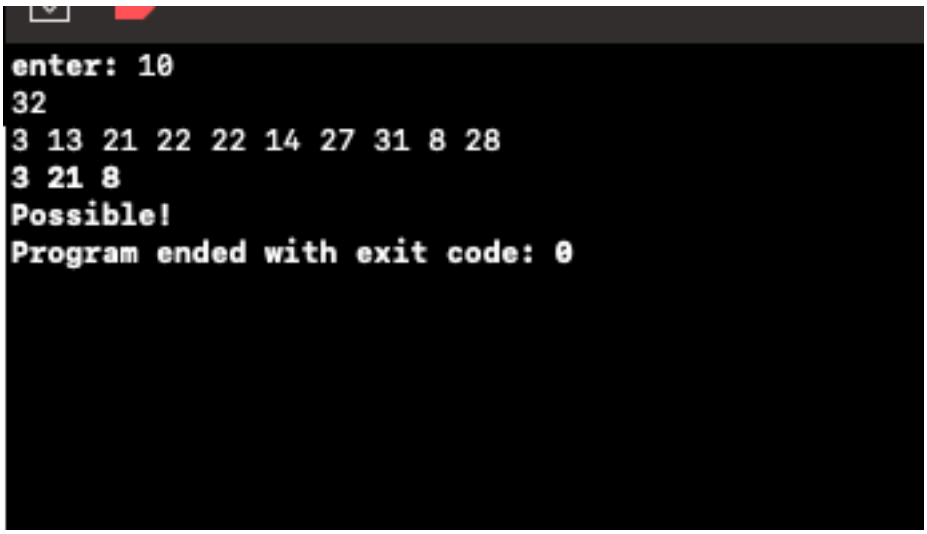
```
Possible!
-- program is finished running --

Enter size: 10

Enter Target: 52

Enter Value
6
24
26
22
7
23
12
7
8
20
6 24 22
Possible!
-- program is finished running --
```

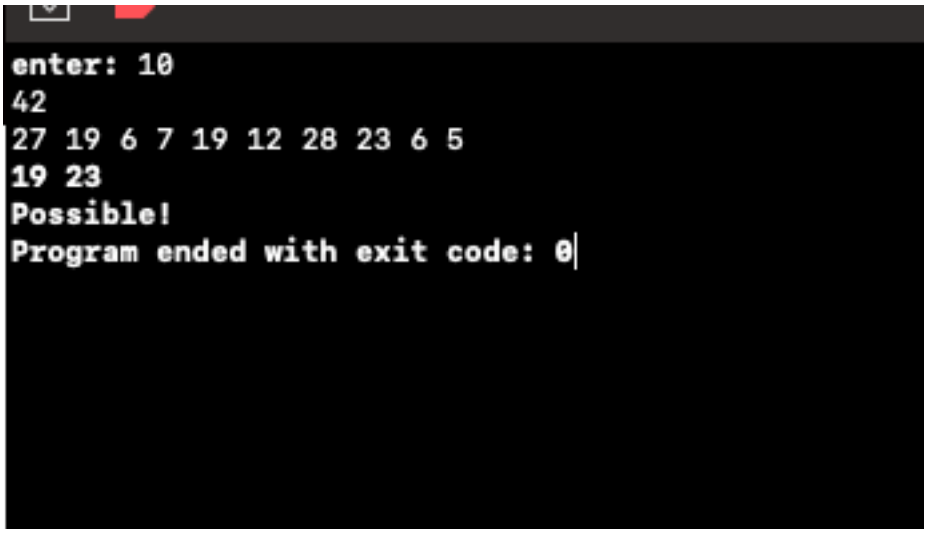
CPP

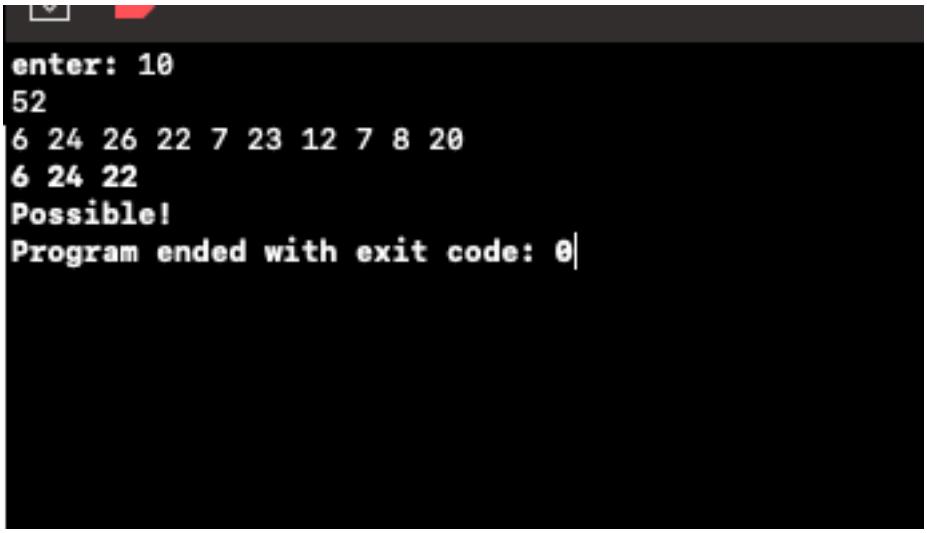

enter: 10
2
9 21 32 2 30 17 28 22 2 12
2
Possible!
Program ended with exit code: 0


enter: 10
12
30 30 17 31 29 30 26 30 10 25
Not possible!
Program ended with exit code: 0


enter: 10
22
19 19 9 22 29 8 31 6 18 30
22
Possible!
Program ended with exit code: 0


enter: 10
32
3 13 21 22 22 14 27 31 8 28
3 21 8
Possible!
Program ended with exit code: 0


enter: 10
42
27 19 6 7 19 12 28 23 6 5
19 23
Possible!
Program ended with exit code: 0


enter: 10
52
6 24 26 22 7 23 12 7 8 20
6 24 22
Possible!
Program ended with exit code: 0